

2015

## Bottom Friction Assessment for Hydrodynamic Currents in the Lower St. Johns River

Henok Kefelegn Demissie  
*University of North Florida, hmyr12@yahoo.com*

Follow this and additional works at: <https://digitalcommons.unf.edu/etd>



Part of the [Civil Engineering Commons](#)

---

### Suggested Citation

Demissie, Henok Kefelegn, "Bottom Friction Assessment for Hydrodynamic Currents in the Lower St. Johns River" (2015). *UNF Graduate Theses and Dissertations*. 570.  
<https://digitalcommons.unf.edu/etd/570>

This Master's Thesis is brought to you for free and open access by the Student Scholarship at UNF Digital Commons. It has been accepted for inclusion in UNF Graduate Theses and Dissertations by an authorized administrator of UNF Digital Commons. For more information, please contact [Digital Projects](#).  
© 2015 All Rights Reserved

Bottom Friction Assessment for Hydrodynamic Currents in the Lower St. Johns River

by

Henok Kefelegn Demissie

A thesis submitted to the Department of Civil Engineering

in partial fulfillment of the requirements for the degree of

Master of Science in Civil Engineering

UNIVERSITY OF NORTH FLORIDA

COLLEGE OF COMPUTING ENGINEERING AND CONSTRUCTION

May, 2015

Bottom Friction Assessment for Hydrodynamic Currents in the Lower St. Johns River is approved:

---

Dr. Peter Bacopoulos, PhD  
Thesis Advisor and Committee Chair

---

Dr. William R. Dally, PhD, P.E  
Committee Member

---

Dr. Nick W. Hudyma, PhD, P.E  
Committee Member

Accepted for the School of Engineering:

---

Dr. Murat M. Tiryakioglu, PhD  
Director of School of Engineering

Accepted for the College of Computing Engineering and Construction:

---

Dr. Mark A. Tumeo, PhD, P.E  
Dean of the College of Computing Engineering and Construction

Accepted for the University:

---

Dr. John Kantner  
Dean of the Graduate School

## DEDICATION

*I dedicate this work and give special thanks to my loving and caring wife Mahider Hailu and my wonderful children Yonatan and Rediham for being there for me.*

*All of you have been my best cheerleaders.*

*A special feeling of gratitude to my mother and father; Alemtsehay Kassaye and Kefelegn*

*Demissie whose words of support and encouragement ring in my ears.*

## ACKNOWLEDGMENTS

This thesis would not have been possible without the guidance and the help of several individuals who, in one way or another, contributed and offered their valuable assistance in the preparation and completion of this study.

First and foremost, my utmost gratitude to Dr. Don Resio, Professor and Director, Taylor Engineering Research Institute (TERI) whose sincerity and generosity I will never forget. I am sure that without Dr. Resio's guidance and support I would have not been a student in Coastal Engineering field and TERI Program.

I would like to thank my committee chair Dr. Peter Bacopoulos, for his unselfish and unfailing support as my thesis adviser as well as the moral support and steadfast encouragement to complete this study. I also thank my committee members Dr. William Dally and Dr. Nick W Hudyma for the guidance and support. A special thank you to Dr. William Dally for providing the data used in the research.

In addition, a thank you to Mrs. Hollis, for being accommodating to my queries and for re-editing the electronic version of this thesis. I also thank Coastal Engineering students who in one way or another were assistance especially Mr. Michael Hunter Bredesen for his support.

This research was funded in part under Award No. NA10OAR4170079 from Florida Sea Grant (FSG) and the "St. Johns River Economic Impact Study: Phase 1 – Ecosystem Services Valuation" project from the St. Johns River Water Management District (SJRWMD).

Last but not the least, my family and the one above all of us, the omnipresent God, for answering my prayers for giving me the strength to plod on despite my constitution wanting to give up and throw in the towel, thank you so much Dear Lord.

## TABLE OF CONTENT

	Page
Dedication .....	iii
Acknowledgments .....	iv
Table of Content .....	v
List of Tables .....	vii
List of Figures .....	viii
Abstract .....	x
Chapter 1: Introduction .....	1
Chapter 2: Observation Data Analysis .....	4
Current Measurements .....	5
Water Surface Elevation .....	16
Freshwater Inflows .....	17
Stream Flow Data .....	19
Chapter 3: Manning's roughness coefficient .....	21
Chezy and Manning Friction Formulations .....	22
Factors Affecting Manning's n .....	26
Directional Manning's n .....	28
CHAPTER 4: Hydrodynamic Modeling .....	34
Shallow Water Equations .....	34
Bottom Friction Formulations in ADCIRC .....	36
ADCIRC Model Setup .....	37
Fort.14 File – Domain Mesh .....	38
Fort.15 File – Parameter File .....	40
Fort.13 File – Nodal Attribute File .....	41
Development of Directional Manning's n .....	44
Validation of the Modified ADCIRC Model .....	50
CHAPTER 5: Results and Discussion .....	53
Hydrodynamic Model Validation .....	55
CHAPTER 6: Conclusions .....	58

References .....	60
Appendix .....	67
Vita .....	106

## LIST OF TABLES

	Page
Table 2.1: Cast station locations .....	7
Table 2.2: ADCP measurements for deployment 1 and 2. AVG is the depth-averaged value of velocity magnitude for each location. STD is the standard deviation of the depth-dependent variability of velocity magnitude for each location .....	15
Table 2.3: General characteristics of the Twenty one tributaries based on flow measurements .....	19
Table 3.1: Factors affecting Manning's roughness coefficient .....	27
Table 3.2: Lists the appropriate values to be used in Eq. 3-20 .....	29
Table 4.1: Model input files (Simulations 1, 2 and 3) .....	38
Table 4.2: Computation of roughness factor $n_o$ .....	45
Table 4.3: Model input files (Simulations 4 and 5) .....	48
Table 5.1: RMSEs and $R^2$ values for Day 1 and 2 simulations .....	54
Table 5.2: RMSEs and $R^2$ values for model validation. ....	56



## LIST OF FIGURES

	Page
Figure 2.1: Lower St. Johns River domain area (Google Earth image).....	5
Figure 2.2a: ADCP measurement locations, ebb tide, 06/17/2009.....	8
Figure 2.2b: ADCP measurement locations, flood tide, 06/17/2009.....	8
Figure 2.3: ADCP measurement velocity speed (Locations 01 to 25) .....	9
Figure 2.4a: Depth averaged velocity magnitude (Locations 01 to 25).....	10
Figure 2.4b: Depth averaged velocity direction (Locations 01 to 25) .....	10
Figure 2.5: ADCP measurement locations, ebb tide, 06/18/2009.....	11
Figure 2.6: Depth averaged velocity magnitude with error bars indicating +/- standard deviations (Locations 01 to 25).....	13
Figure 2.7: Depth averaged velocity magnitude with error bars indicating +/- standard deviations (Locations 26 to 43) .....	13
Figure 2.8: Location of fixed tide gauge instrument – Dames Point (Google Earth).....	16
Figure 2.9: Plot of barometric pressure.....	17
Figure 2.10: Pressure data converted to tide elevation (m) relative to NAVD88.....	17
Figure 2.11: Locations of watershed (Bacopoulos, 2015) .....	18
Figure 2.12: Flow discharge at Acosta Bridge.....	20
Figure 3.1: The control volume from (Chow, 1959) .....	22
Figure 3.2: Ebb and flood tide flow .....	30
Figure 4.1: Finite element mesh for the Western North Atlantic Tidal model domain: OCEAN (Google Earth).....	39

Figure 4.2: ADCIRC mesh triangulation (left) with colored elevation (right) .....	40
Figure 4.3: Simulations 1, 2 and 3, 17-Jun-2009 .....	43
Figure 4.4: Simulations 1, 2 and 3, 18-Jun-2009 .....	43
Figure 4.5: Modified subroutine (Apply2DBottomFriction) .....	49
Figure 4.6: Roughness factor contours for different sample flow directions. (Note: Ebb and flood direction are shown in arrows for reference only, they tend to adjust with the orientation of the river channel.) .....	50
Figure 4.7: Simulations 1 to 5, Cast 01 to 25 .....	51
Figure 4.8: Simulations 1 to 5, Cast 26 to 43 .....	52
Figure 5.1: Validation plot, Acosta Bridge stream flow .....	55
Figure 5.2: Validation plot, Dames Point Bridge water elevation .....	56

## ABSTRACT

A preexisting version of the 2D finite-element hydrodynamic model (code) ADCIRC was modified to enable assimilation of velocity data for calculation of longwave hydrodynamics of the lower St. Johns River. The data assimilation also enables model calibration and parameter estimation of directionally variant Manning's  $n$  value using an anisotropic formulation of bottom roughness. This modified version of the ADCIRC code differs from the original ADCIRC model, as it introduces a module to provide evaluation of directional Manning's roughness coefficient using observed velocity data for ebb and flood flow durations. The vector-based directional Manning's  $n$  value is found by comparing the observed velocity data with the ADCIRC output from the original model dataset, depicting how the friction factor depends on flow direction. The modified ADCIRC model was calibrated for the velocity dataset and then validated with a different dataset of water surface elevation and streamflow. It is shown that the influence of the directional variability in bottom roughness is a significant factor in calibration of the model, especially given the to-and-fro nature of the tidal motions, which is contrary to present practices that ignore the temporal variability and any anisotropy in bottom roughness. This thesis makes a measured impact on how 2D hydrodynamic models (herein demonstrated with ADCIRC) are able to represent the directionality of bottom roughness in hydrodynamic simulation.

## CHAPTER 1: INTRODUCTION

The role of bottom friction in modeling the tidal dynamics of shallow seas and coastal rivers is well appreciated and relatively well studied (at least for vertically integrated models). However, despite considerable theoretical, experimental, and numerical research, the representation and parameterization of frictional processes remains an uncertain feature of modeling. To date, Manning's roughness has been described with a constant friction coefficient in time, independent of flow direction. The spatially variable Manning's  $n$  has been used in practically all of the hydrodynamic models. These approaches assume that Manning's  $n$  is a numerical constant. Yet, adjustment of bottom friction parameter(s) remains the primary means of calibration for most hydrodynamic models. The objective of the study is to develop a bottom friction formulation for tidal simulation that considers the directional variability in bottom roughness.

In coastal waters, tides are generally classified as astronomical tides, compound tides, or overtides. Astronomical tides result from the gravitational forces exerted by the sun and the moon. Compound tides and overtides arise from the nonlinear interactions between constituents; they are often called shallow water tides because nonlinear phenomena (e.g., bottom friction) generally become important in shallow regions. Importantly, tides are periodic in nature and promote directionally variant motions. Generally, tidal currents become stronger near the coast and play an increasingly important role in local circulation. Because of the rotating nature of the tide wave in many locations (especially inland seas and enclosed basins), ebb and flood currents follow different paths (i.e., are in different directions).

Because of complicated shoreline configuration and bottom bathymetric gradients, bottom friction is usually parameterized by spatially varying friction factors. However, these friction factors are assumed to be constant (i.e., not changing with time). Observation data suggest that the friction factors may contain some temporal variability. The roughness length of sea bottom and the roughness coefficient vary spatially and temporally if the type and sizes of roughness elements of the sea/river bottom are variable with respect to space and time. A physical explanation for modifying the friction factors (and thus the bottom stress) is that the modifications account for the presence of a steady background current or residual turbulence field which contributes to the frictional processes.

It is often suggested that energy dissipation, especially dissipation caused by bottom friction, depends strongly on topography (Doos et al., 2004). Basically there are two ways to parameterize bottom friction: linear and nonlinear. The linear parameterization is done by a bottom friction coefficient (BFC) multiplied by the velocity, while the nonlinear parameterization calculates bottom stress as a function of a BFC multiplied by the square of the velocity. This detail of the nonlinear parameterization is important because of the interaction between BFC and velocity, i.e., how the velocity depends on the BFC and how the BFC depends on the velocity (magnitude). A major breakthrough this thesis provides to the parameterization of bottom roughness is to redevelop the nonlinear parameterization so that BFC depends on the velocity in its entirety, i.e., both magnitude and direction.

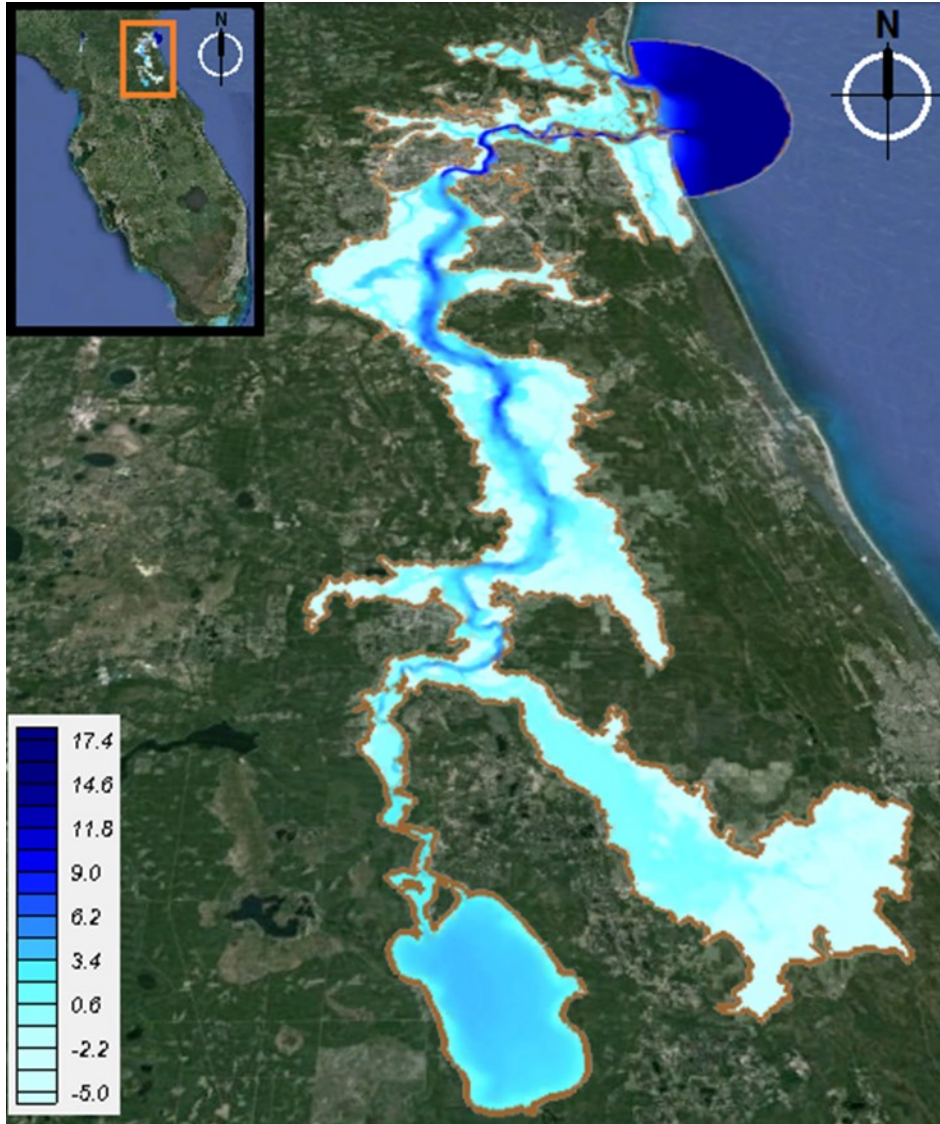
This study details the redevelopment of the bottom friction formulation used within the ADCIRC model to account for directionally variant Manning's  $n$ . The redeveloped bottom friction formulation was demonstrated to perform optimally in a validation of hydrodynamic

simulation for the lower St. Johns River. For this study, the widely used ADCIRC model was used to discretize the lower St. Johns River (LSJR) with a triangular finite element mesh. The mesh is applicable not only for solving the equations of motion, but it also defines the surface topography and the frictional characteristics of the region. Therefore, parameters that describe frictional resistance to water can be defined on the same spatial scale at which the equations are being solved. The physical processes in the ADCIRC hydrodynamic model are described by the depth-integrated shallow water equations. These equations are widely used to describe coupled storm surge, tides, and riverine flows in the coastal ocean and adjacent floodplain. Processes that exist at the physical boundaries of the water column are parameterized, including bottom shear stress. In this thesis, bottom stress was parameterized with the standard Manning's  $n$  coefficient and with a modified directional Manning's  $n$  coefficient based on velocity direction of the flow. Bottom stress was analytically computed via data assimilation of current profile measurements and comparison to model results to find a relationship of the Manning's  $n$  with the direction of the flow, where this relationship informed the bottom friction formulation on how to treat the directionality of Manning's  $n$ , given the to-and-fro nature of ebb and flood tides in the lower St. Johns River.

## CHAPTER 2: OBSERVATION DATA ANALYSIS

The study location is the lower St. Johns River (LSJR) in Florida. The St. Johns River is the longest river in the state of Florida and it is significant for commercial and recreational use. At 500 km long, it winds through or borders twelve counties, three of which are the state's largest. The drop in elevation from the headwaters to the mouth is less than 9 meters (Toth, 1993). Like most Florida waterways, the St. Johns has a very low flow rate, 0.13 m/s, and is often described as "lazy." The full extent of the St. Johns River watershed encompasses over 22,000 square km (Sucsy, 2002). The LSJR basin area is approximately 6,700 square km (Figure 2.1). The tributaries of the LSJR are varied both in size and water type. Twenty-one LSJR basin tributaries were selected for inclusion in this study. Their watersheds range in size from about 61 square km to 1478 square km. A number of small, short streams with limited drainage areas also drain directly to the river.

Currents in the LSJR are tidally dominated. Because the river is basically a constricted channel, the currents are rectilinear (or reversing), in that the water flows alternately in approximately opposite directions, with a slack water at each reversal of direction consisting of two flood and two ebb periods each day. In addition to tides, local winds and freshwater river inflows can influence water motions in the LSJR; however, at times, remote wind impacts can be a major contributor to the overall water level in the LSJR (Bacopoulos, 2009). At the river entrance, the maximum flood and ebb currents occur approximately one hour before the high and low tides at the river entrance (NOAA, 1999).



*Figure 2.1. Lower St. Johns River domain area (Google Earth image).*

#### Current Measurements

Different datasets of water levels and flow velocities have been collected by Surfbreak Engineering Sciences, Inc, (U.S. Army Corps of Engineers, USACE, 2009). Surfbreak Engineering collected the data for the USACE Jacksonville District (SAJ) in order to conduct a feasibility study of modifications of Jacksonville Harbor, with the goal of improving ship navigation in the LSJR and in the harbor. These datasets are used for the data assimilation



performed in this study for the development of the directionally variant bottom friction formulation. Data were collected using both fixed and mobile instruments. Table 2.1 summarizes the current measurements used for the study.

The measurement included current profiles using a boat-mounted Acoustic Doppler Current Profiler (ADCP). The current profile data were collected during both ebb and flood durations. As a result, these data provide a basis for assessing the directionality of bottom roughness with regards to the rectilinear tidal motions occurring in the LSJR. Vertical profiles of current speed and direction were measured from the casting boat while at anchor, using a down-looking ADCP, and the measurement pattern as dictated in Figures 2.2a, 2.2b, 2.4a and 2.4b. Details of the mobile casting work, as executed, are provided in Table 2.1. The sampling time interval for data collection was 1 minute. The sampling depth interval for data collection was 0.5 m. Figures 2.3 and 2.5 show the depth-dependent contour plots for deployment 1 and 2 respectively. The velocity data were collected for approximately 7 minutes at each location, while there were 5–10 minutes of non-sampling time in between locations, which is why there appears to be columns of data with gaps in between each column (Figures 2.3 and 2.5). The gap within each column of data represents the measurement of the local channel bottom. The velocity measurements below the gaps within the columns were not considered in the analysis, since they do not reflect the actual condition. Velocity speeds range from 0 to 1.4 m/s. Velocity directions range from 0 to 360 degrees. In all, measurements at 43 locations were made, and an ADCP file accompanied each measurement. The position of the boat during each cast is shown in Table 2.1.

Table 2.1

*Cast Station Locations*

Cast Number	Lat/Log	Date	Time	Flow Condition
1	N30°024.147',W81°24.099'	6/17/2009	7:22 AM	Ebb
2	N30°024.113',W81°25.086'	6/17/2009	7:45 AM	Ebb
3	N30°023.917',W81°25.959'	6/17/2009	8:13 AM	Ebb
4	N30°023.046',W81°25.549'	6/17/2009	8:33 AM	Ebb
5	N30°023.022',W81°27.325'	6/17/2009	9:02 AM	Ebb
6	N30°023.489',W81°29.733'	6/17/2009	9:27 AM	Ebb
7	N30°023.413',W81°31.737'	6/17/2009	9:46 AM	Ebb
8	N30°023.032',W81°33.701'	6/17/2009	10:04 AM	Ebb
9	N30°024.509',W81°35.271'	6/17/2009	10:24 AM	Ebb
10	N30°023.950',W81°37.189'	6/17/2009	10:43 AM	Ebb
11	N30°022.338',W81°37.617'	6/17/2009	11:03 AM	Ebb
12	N30°030.797',W81°37.156'	6/17/2009	11:20 AM	Ebb
13	N30°018.897',W81°37.818'	6/17/2009	11:40 AM	Ebb
14	N30°024.139',W81°24.158'	6/17/2009	2:10 PM	Flood
15	N30°024.110',W81°25.141'	6/17/2009	2:30 PM	Flood
16	N30°023.791',W81°26.039'	6/17/2009	3:13 PM	Flood
17	N30°022.987',W81°26.593'	6/17/2009	3:28 PM	Flood
18	N30°022.996',W81°27.297'	6/17/2009	3:49 PM	Flood
19	N30°023.595',W81°28.602'	6/17/2009	4:11 PM	Flood
20	N30°023.458',W81°29.852'	6/17/2009	4:27 PM	Flood
21	N30°023.434',W81°30.688'	6/17/2009	4:44 PM	Flood
22	N30°023.077',W81°33.034'	6/17/2009	5:03 PM	Flood
23	N30°023.042',W81°33.763'	6/17/2009	5:22 PM	Flood
24	N30°023.817',W81°34.420'	6/17/2009	5:39 PM	Flood
25	N30°023.340',W81°31.737'	6/17/2009	6:02 PM	Flood
26	N30°023.561',W81°28.486'	6/18/2009	9:30 AM	Ebb
27	N30°023.558',W81°29.050'	6/18/2009	9:48 AM	Ebb
28	N30°023.428',W81°30.672'	6/18/2009	10:08 AM	Ebb
29	N30°023.053',W81°33.049'	6/18/2009	10:29 AM	Ebb
30	N30°023.791',W81°34.392'	6/18/2009	10:50 AM	Ebb
31	N30°024.384',W81°36.147'	6/18/2009	11:13 AM	Ebb
32	N30°023.024',W81°37.609'	6/18/2009	11:34 AM	Ebb
33	N30°021.531',W81°36.990'	6/18/2009	11:58 AM	Ebb
34	N30°019.293',W81°37.454'	6/18/2009	12:19 PM	Ebb
35	N30°024.445',W81°35.185'	6/18/2009	3:00 PM	Flood
36	N30°024.379',W81°36.139'	6/18/2009	3:21 PM	Flood
37	N30°023.830',W81°37.191'	6/18/2009	3:40 PM	Flood
38	N30°023.015',W81°37.592'	6/18/2009	4:00 PM	Flood
39	N30°022.328',W81°37.589'	6/18/2009	4:26 PM	Flood
40	N30°021.573',W81°36.967'	6/18/2009	5:03 PM	Flood
41	N30°020.798',W81°37.138'	6/18/2009	5:22 PM	Flood
42	N30°019.281',W81°37.459'	6/18/2009	5:42 PM	Flood
43	N30°018.902',W81°37.836'	6/18/2009	5:58 PM	Flood

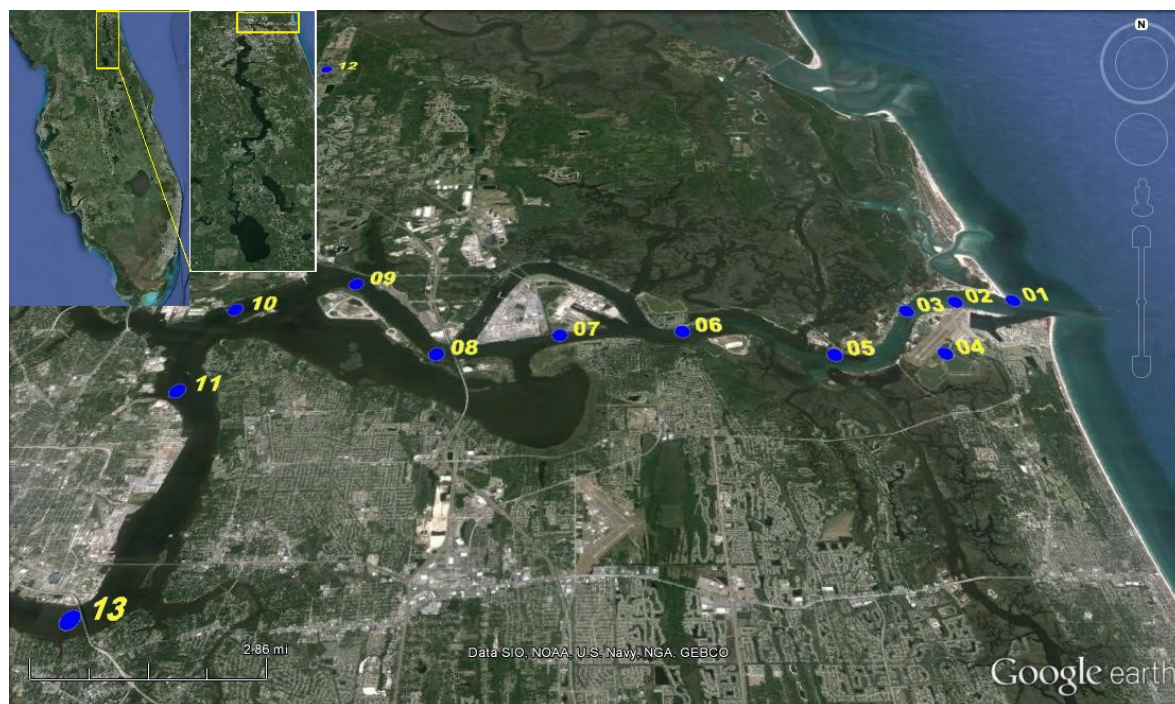


Figure 2.2a. ADCP measurement locations, ebb tide on date 06/17/2009.

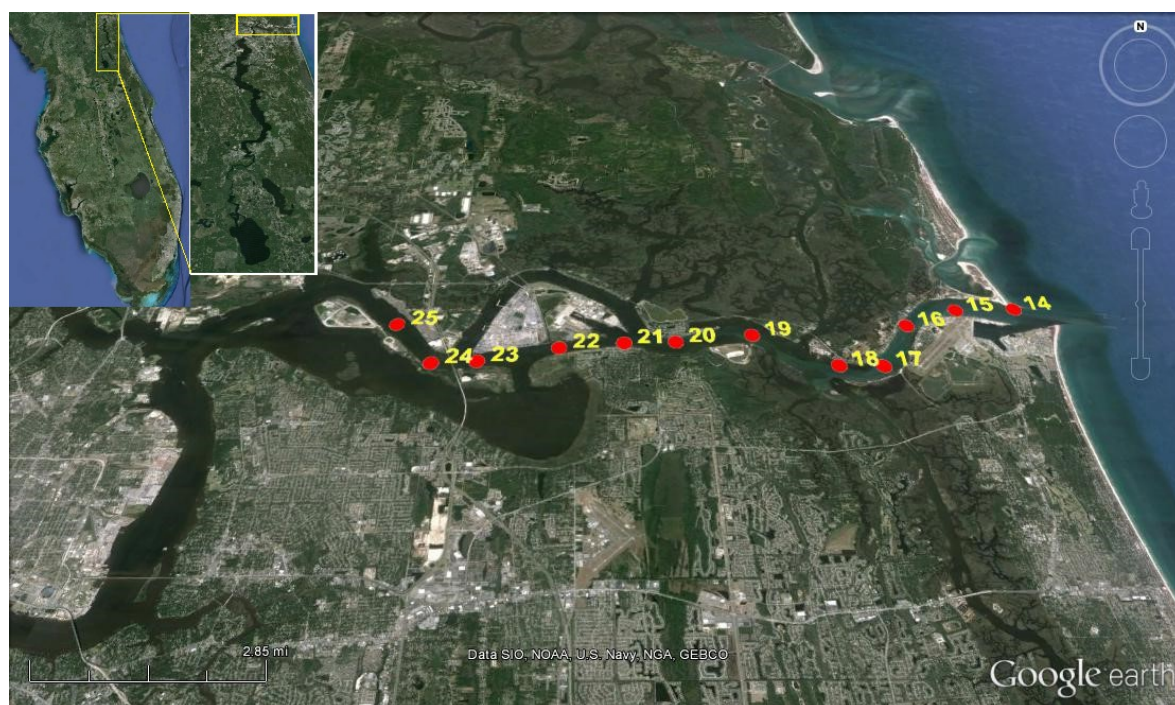


Figure 2.2b. ADCP measurement locations, flood tide on date 06/17/2009.

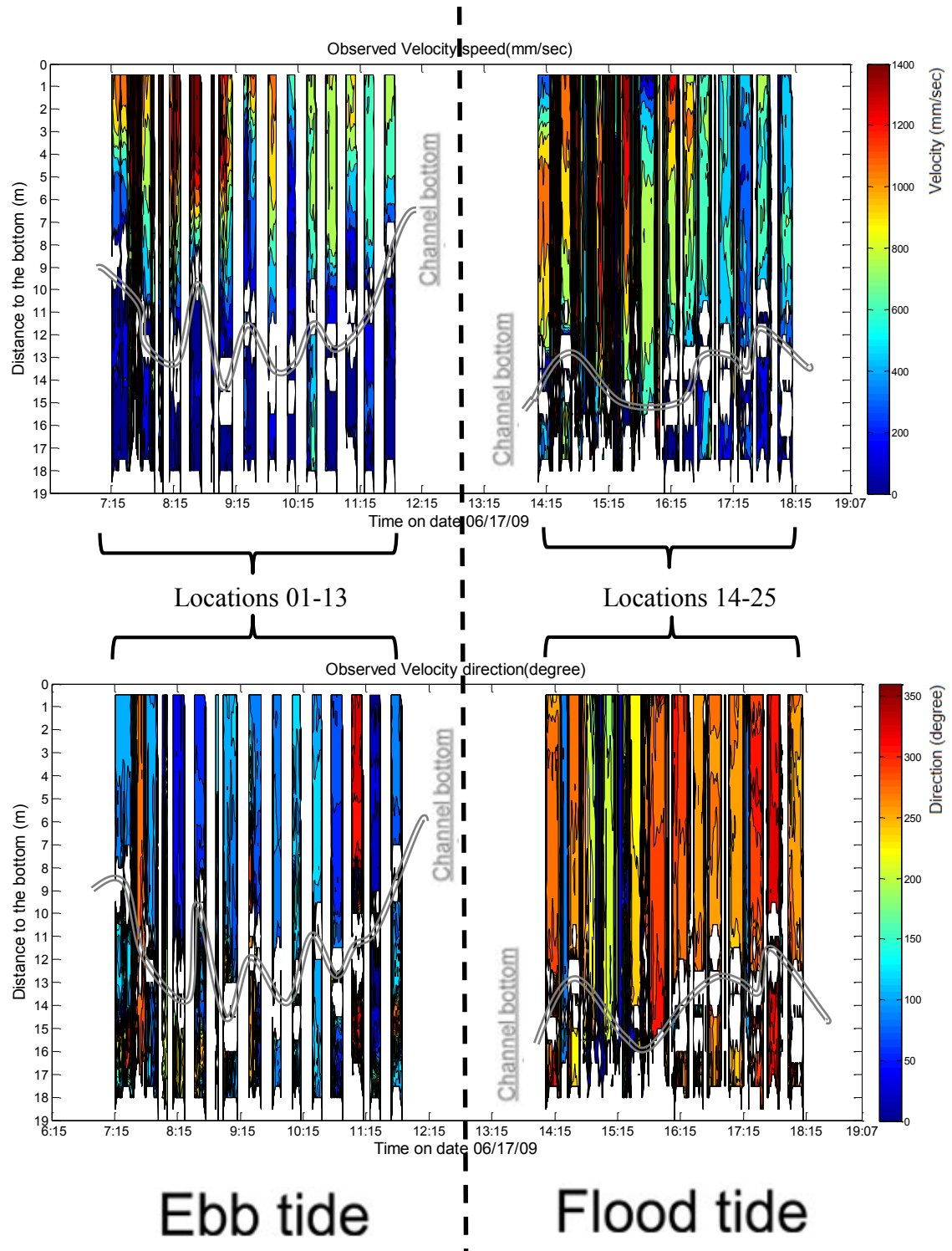


Figure 2.3. ADCP measurement velocity direction (Locations 01 to 25).



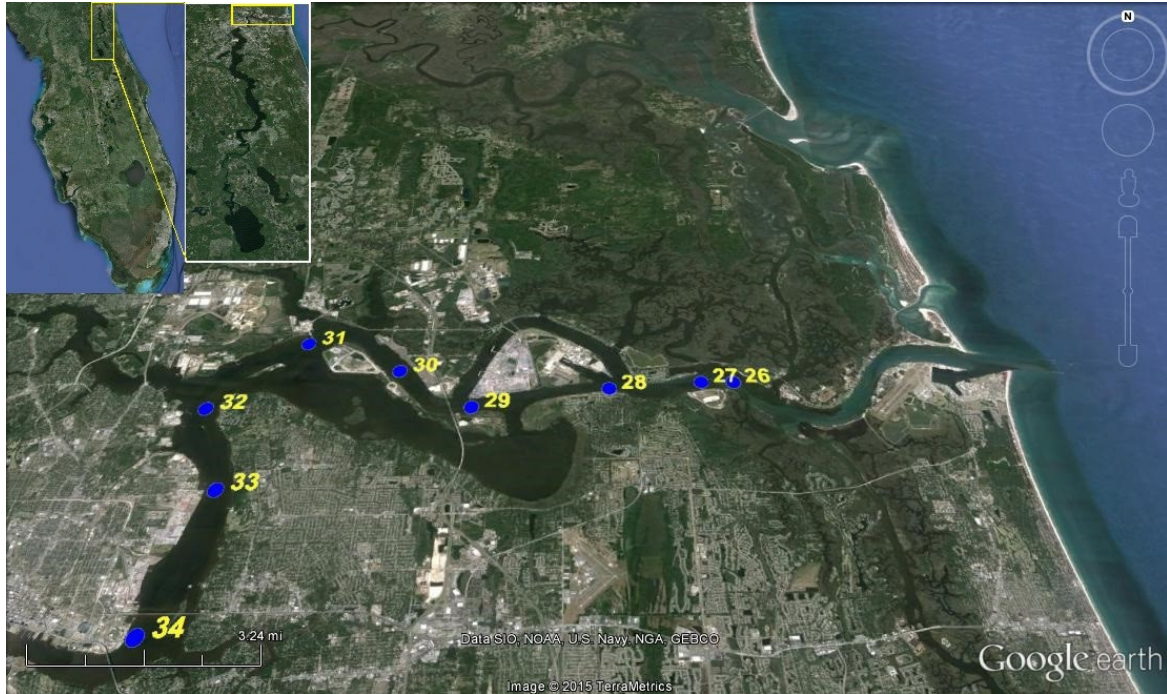


Figure 2.4a. ADCP measurement locations, ebb tide on date 06/17/2009.



Figure 2.4b. ADCP measurement locations, flood tide on date 06/18/2009.

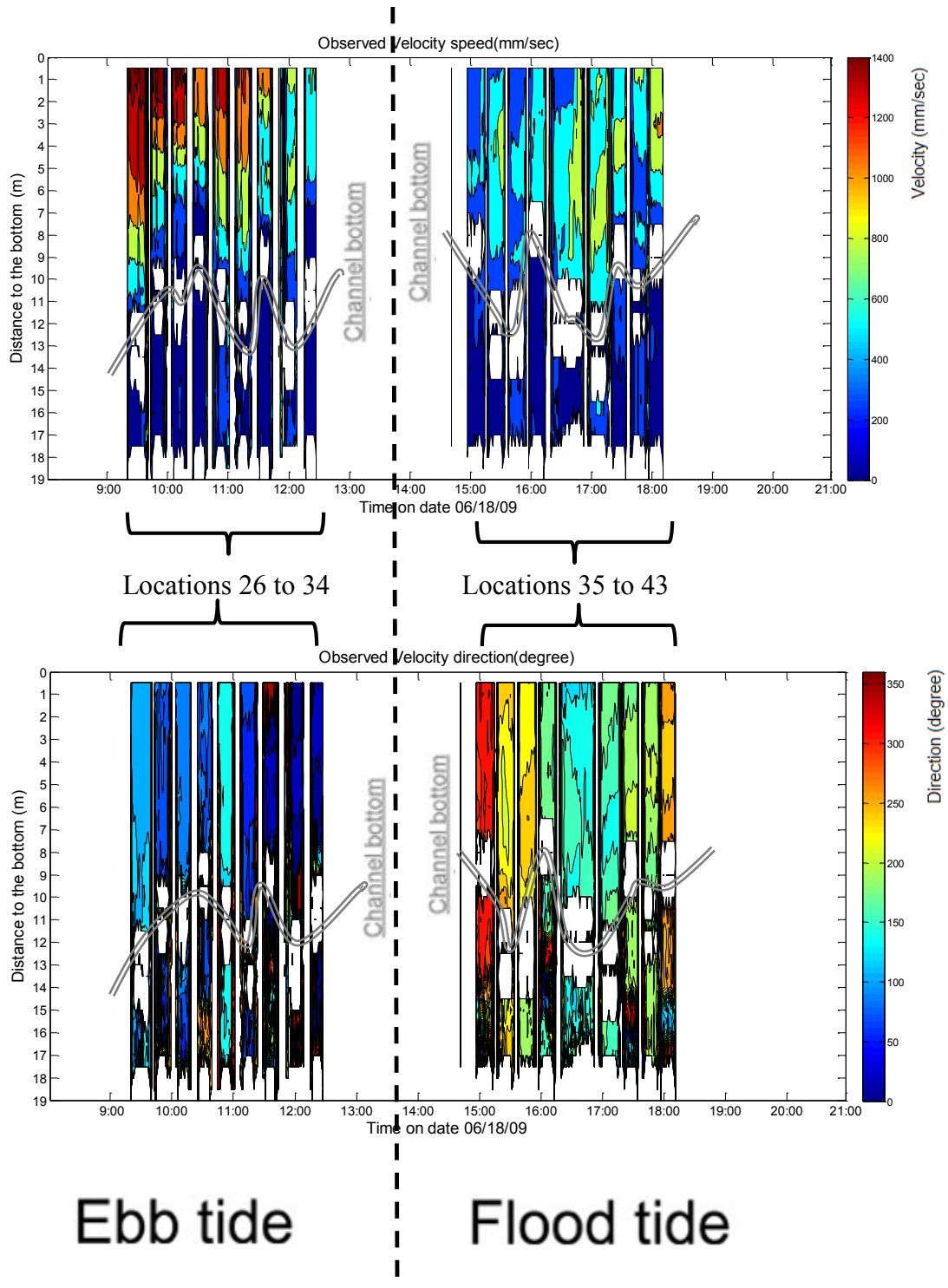


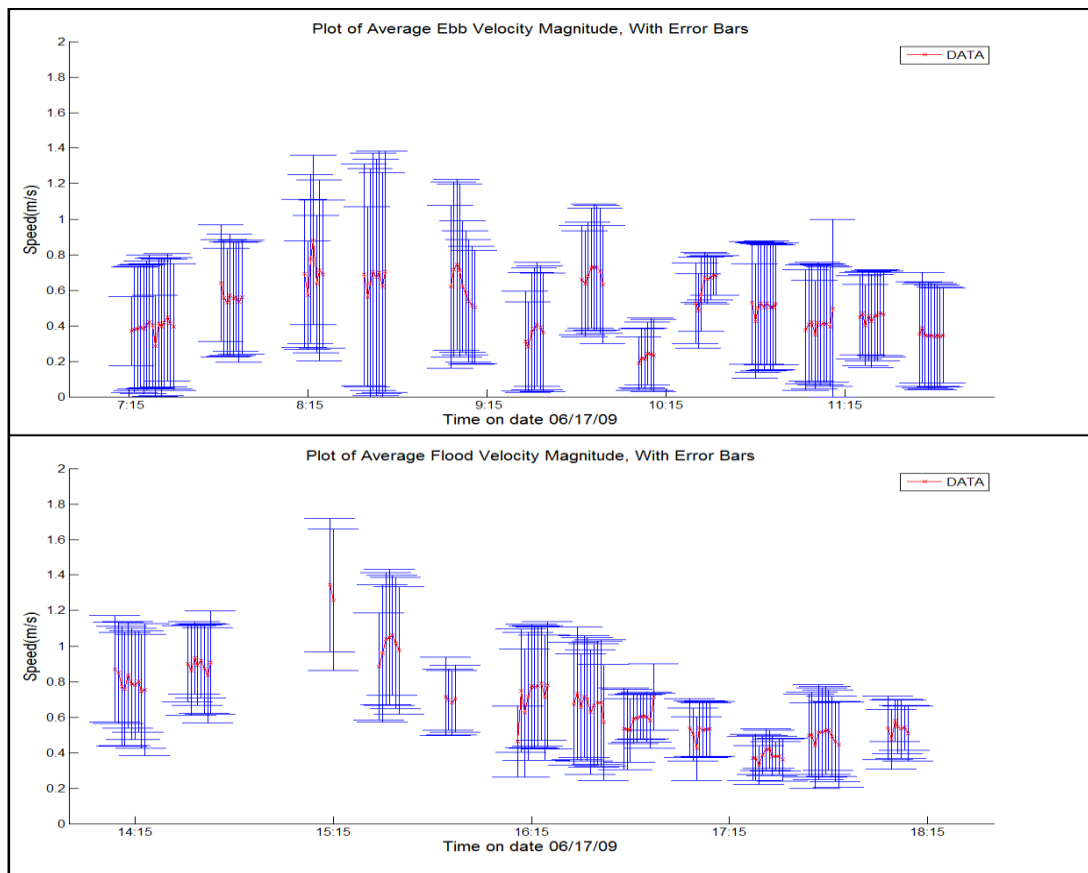
Figure 2.5. ADCP measurement velocity direction (Locations 26 to 43).

At each measurement location, the data show that the velocity direction is oriented essentially in the same direction through the entire water column. On the other hand, there is discernible vertical structure of the velocity speed, where the vertical structure is such that velocity magnitudes are largest at and near the water surface (approximately the first 1–5 m of the vertical water column, depending on location) and decrease fairly linearly to the river bottom (varying from 8 to 18 m deep among the 43 locations).

Velocity data were plotted for the 43 locations and plotted in a depth average manner with error bars indicating  $\pm$  standard deviations (Figure 2.6 for deployment 1 and Figure 2.7 for deployment 2). Statistics were calculated for average (AVG) and standard deviation (STD) of velocity magnitude for each of the 43 locations (Tables 2.2). The overall AVG and STD values for ebb duration of deployment 1 were computed to be 0.50 and 0.39 m/s, respectively, which corresponds to an overall AVG:STD value of 78%. For ebb duration of deployment 2, overall AVG, STD and AVG:STD values were computed to be 0.60 m/s, 0.48 m/s and 80%, respectively. The overall AVG, STD and AVG:STD values are similar for ebb durations of deployments 1 and 2, which is because locations 1–13 and locations 26–34 are fairly collocated with one another (Figures 2.2 and 2.4). Nonetheless, the data for ebb durations show depth-averaged currents with speeds of 0.39–0.94 m/s near the river mouth (refer to locations 01–03 and locations 26–28) and with speeds of 0.35–0.59 m/s near downtown Jacksonville (river km 30) (refer to locations 11–13 and locations 32–34), which is representative of the tidal damping that occurs in lower reaches of the St. Johns River (Sucsy and Morris, 2002).

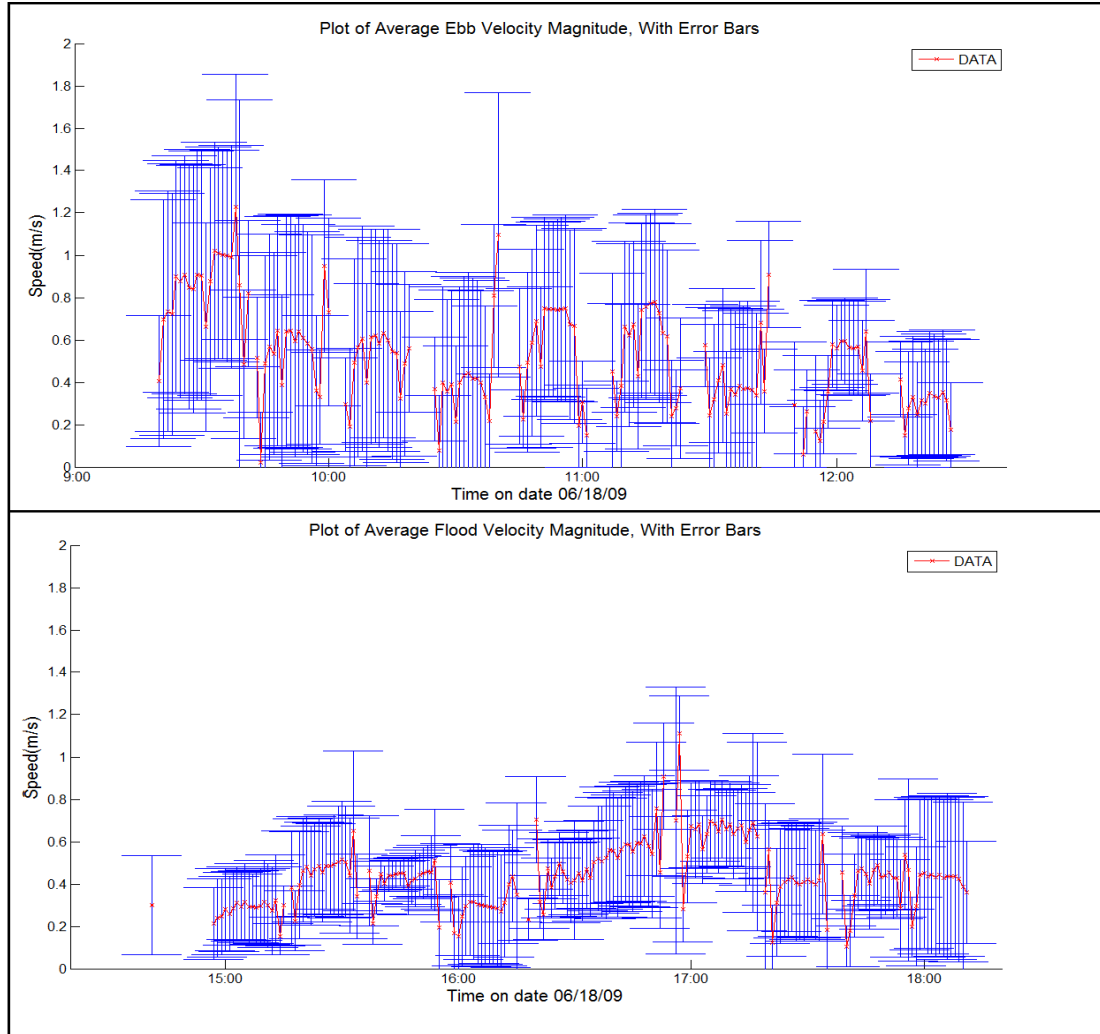
The overall AVG and STD values for flood duration of deployment 1 were computed to be 0.54 and 0.51 m/s, respectively, which corresponds to an overall AVG:STD

value of 94% (Tables 2.2). For flood duration of deployment 2, overall AVG, STD and AVG:STD values were computed to be 0.54 m/s, 0.41 m/s and 76%, respectively. The overall AVG, STD and AVG:STD values are different for flood durations of deployments 1 and 2, which is because locations 14–25 and locations 35–43 are in sequence along the lower 30 km of the St. Johns River (Figures 2.2 and 2.4). Nonetheless, the data for flood durations show depth-averaged currents with speeds of 0.79–1.30 m/s near the river mouth (refer to locations 14–16) and with speeds of 0.43–0.46 m/s near downtown Jacksonville (river km 30) (refer to locations 41–43), which is representative of the tidal damping that occurs in lower reaches of the St. Johns River (Sucsy and Morris, 2002).



*Figure 2.6.* Depth averaged velocity magnitude with error bars indicating  $\pm$  standard deviations (Locations 01 to 25).





*Figure 2.7.* Depth averaged velocity magnitude with error bars indicating  $\pm$  standard deviations (Locations 26 to 43).

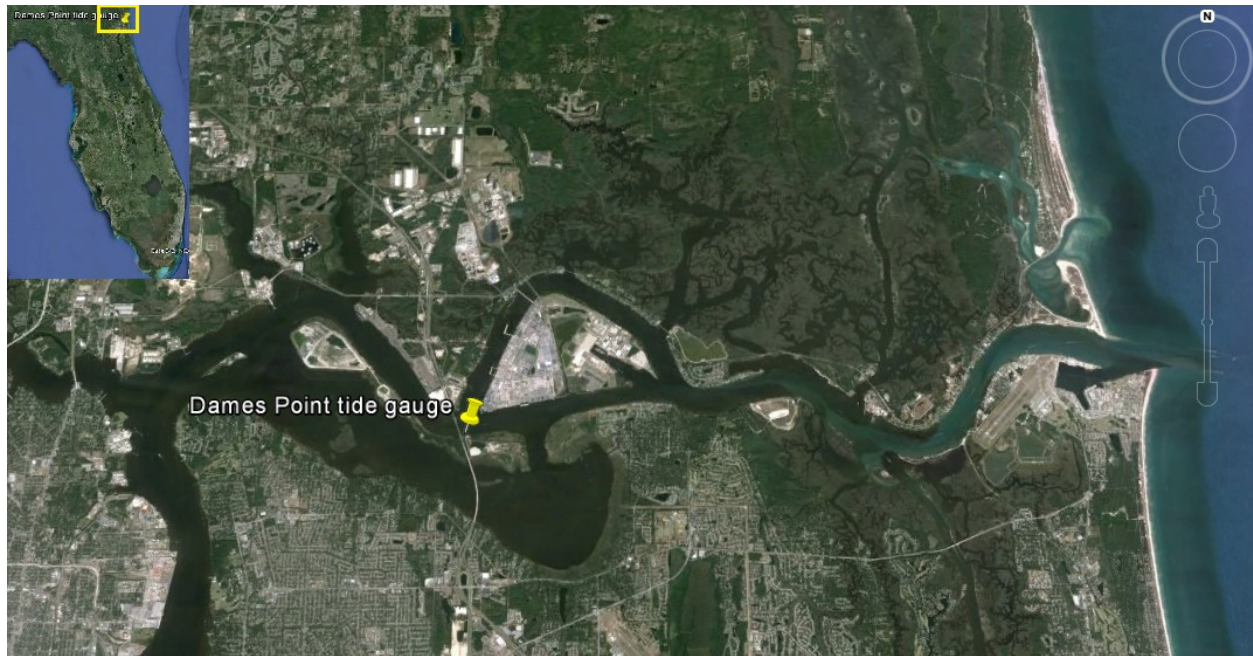
Table 2.2

*ADCP measurements for deployment 1 and 2. AVG is the depth-averaged value of velocity magnitude for each location. STD is the standard deviation of the depth-dependent variability of velocity magnitude for each location.*

Deployment 1, June 17, 2009					Deployment 2, June 18, 2009				
No	Tide	Speed			No.	Tide	Speed		
		AVG (m/s)	STD (m/s)	AVG:STD (%)			AVG (m/s)	STD (m/s)	AVG:STD (%)
1	Ebb	0.39	0.35	89	26	Ebb	0.94	0.63	67
2	Ebb	0.56	0.33	58	27	Ebb	0.61	0.48	79
3	Ebb	0.70	0.44	62	28	Ebb	0.54	0.47	87
4	Ebb	0.66	0.64	97	29	Ebb	0.48	0.45	94
5	Ebb	0.62	0.42	68	30	Ebb	0.61	0.40	66
6	Ebb	0.36	0.32	91	31	Ebb	0.62	0.40	65
7	Ebb	0.68	0.33	49	32	Ebb	0.48	0.37	77
8	Ebb	0.22	0.18	81	33	Ebb	0.59	0.39	66
9	Ebb	0.62	0.18	30	34	Ebb	0.35	0.26	75
10	Ebb	0.50	0.35	69	OVERALL		0.60	0.48	80
11	Ebb	0.41	0.36	87	35	Flood	0.32	0.29	90
12	Ebb	0.45	0.25	55	36	Flood	0.48	0.26	53
13	Ebb	0.35	0.29	84	37	Flood	0.45	0.15	35
OVERALL		0.50	0.39	78	38	Flood	0.62	0.36	78
14	Flood	0.79	0.32	40	39	Flood	0.55	0.34	62
15	Flood	0.89	0.24	27	40	Flood	0.75	0.40	53
16	Flood	1.30	0.39	30	41	Flood	0.46	0.29	64
17	Flood	1.00	0.36	37	42	Flood	0.43	0.23	54
18	Flood	0.70	0.19	28	43	Flood	0.46	0.36	79
19	Flood	0.71	0.35	49	OVERALL		0.54	0.41	76
20	Flood	0.67	0.35	52					
21	Flood	0.59	0.18	30					
22	Flood	0.51	0.17	32					
23	Flood	0.38	0.11	30					
24	Flood	0.49	0.24	49					
25	Flood	0.53	0.16	30					
OVERALL		0.54	0.51	94					

### Water Surface Elevation

A fixed instrument was used to measure water elevation using a Level TROLL® 100 self-recording pressure based tide gauge near Dames Point (Figure 2.8). Measurements of atmospheric pressure were also taken using a HOBO Micro Station Self-recording barometer manufactured by the Onset Computer Corporation. The location of the barometer was N30° 18.897', W81° 27.990'. The processed data are plotted in Figure 2.9. The pressure data were converted to water elevation by subtracting the barometric pressure from the water level data as measured by the tide gauge. Figure 2.10 shows the water elevation plot relative to NAVD88. The water surface elevation data are used for hydrodynamic model validation purpose in Chapter 5.



*Figure 2.8.* Location of fixed tide gauge instrument – Dames Point (Google Earth).

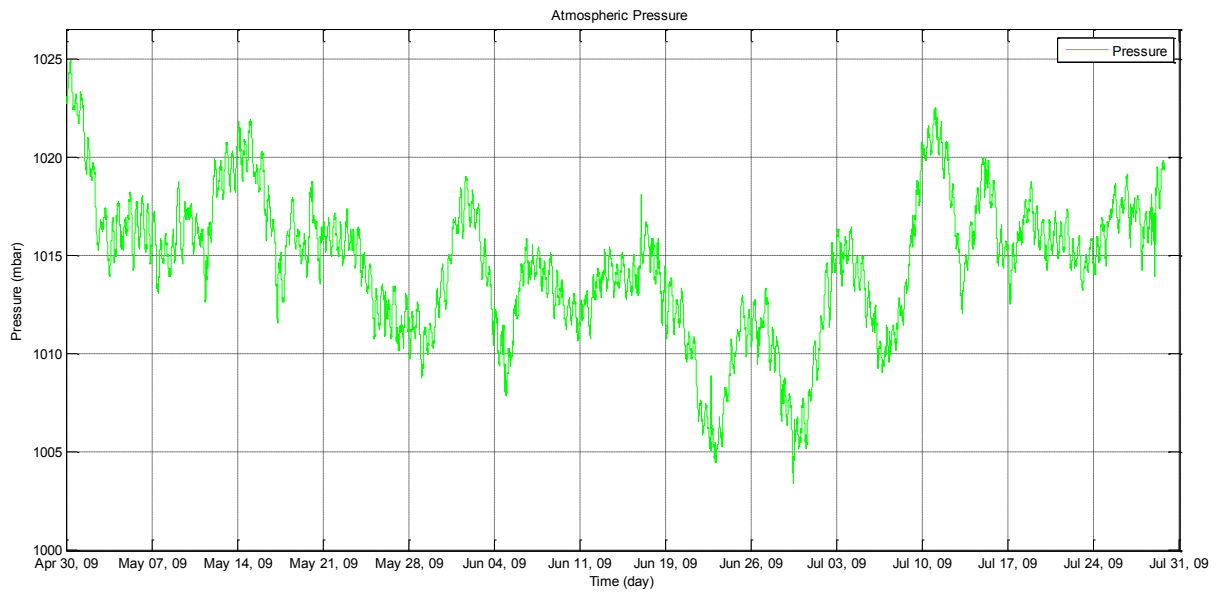


Figure 2.9. Plot of barometric pressure.

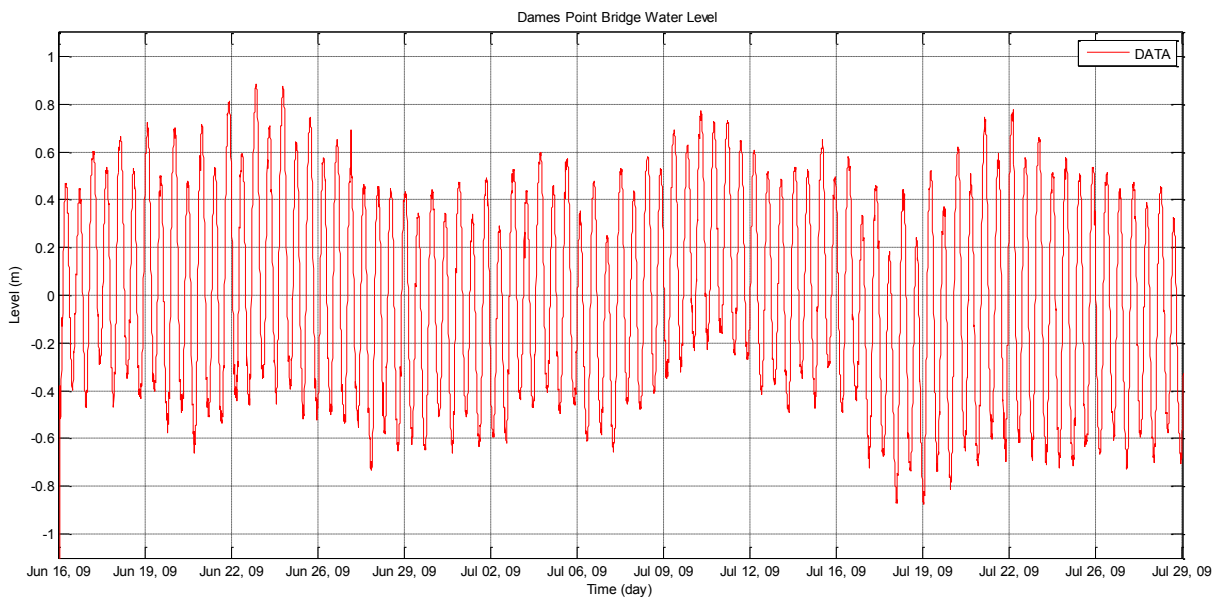


Figure 2.10. Pressure data converted to tide elevation (m) relative to NAVD88.

### Freshwater Inflows

The magnitude of flow in the river is a measure of the rate at which volumes of water move. Flow distributions are needed for describing the flow condition. The total flow in the river is the volume of water moving at a cross section in the river over a period of time (units of

volume divided by time). The flow generally increases proceeding downstream as tributaries join the main stem. Flow can be treated as an instantaneous quantity, but more often it is expressed as a time averaged quantity.

In general, the movement of water in the river varies in all directions longitudinally, laterally, and/or vertically in response to local forces. The predominant forces determining the total flow in the St. Johns River are the volume of water stored upstream as a result of previous season activity, the flow caused by tidal forces, the amount of inflow from tributaries, the amount of direct rainfall, and the effect of wind.

The flow in the river depends on the amount of water in storage and the relative magnitudes of inflows, outflows, and climatological forces (primarily wind and pressure). When the net freshwater inflow is positive, the duration and volume of downstream flows in the LSJR tend to increase, while the duration and volume of upstream flows tend to decrease (Anderson and Goolsby, 1973).

Graphic redacted,  
paper copy available  
upon request to home  
institution.

Table 2.3

*General Characteristics of the 21 Tributaries Based on Flow Measurements*

Tributary basin	Basin Area (km <sup>2</sup> )	Maximum flow (m <sup>3</sup> /s)	Minimum flow (m <sup>3</sup> /s)	Average flow (m <sup>3</sup> /s)
0	1097.35	114.59	4.54	19.91
1	589.01	51.53	1.67	30.02
2	33.62	15.57	0.97	4.21
3	66.03	34.67	1.47	11.56
4	82.16	26.88	1.75	14.01
5	119.47	78.27	3.33	27.98
6	107.82	59.90	3.35	29.05
7	140.22	140.20	4.41	51.10
8	135.34	227.32	3.89	50.82
9	198.18	155.69	4.17	39.04
10	800.40	266.31	2.53	82.26
11	774.87	252.27	2.47	76.84
12	281.50	281.84	0.87	33.74
13	63.50	148.98	0.69	21.51
14	100.04	193.13	1.12	31.20
15	171.81	243.46	1.33	47.76
16	91.74	127.56	0.70	23.09
17	78.61	120.15	0.76	18.13
18	75.82	134.12	0.56	21.09
19	9891.97	186.98	15.64	54.23
20	2023.41	28.36	0.15	1.61

## Stream Flow Data

Ninety-two days of flow data from USGS were analyzed for the duration of May 01 to August 01, 2009 at the Acosta Bridge station 2246500 (Figure 2.12). The location of USGS station is at latitude 30°19'20", longitude 81°39'56" referenced to North American Datum of 1927, Duval County, Florida, and Hydrologic Unit 03080103. It is near the center of the channel under the Acosta Bridge at Jacksonville, 2.6 mi upstream from Arlington River, and 23.0 mi upstream from mouth. The instrument used to measure the discharge is a water-stage recorder, acoustic velocity meter. It was found that 56% of the flow is Ebb for the 3 month duration. The stream flow data are used for hydrodynamic model validation purpose in Chapter 5.

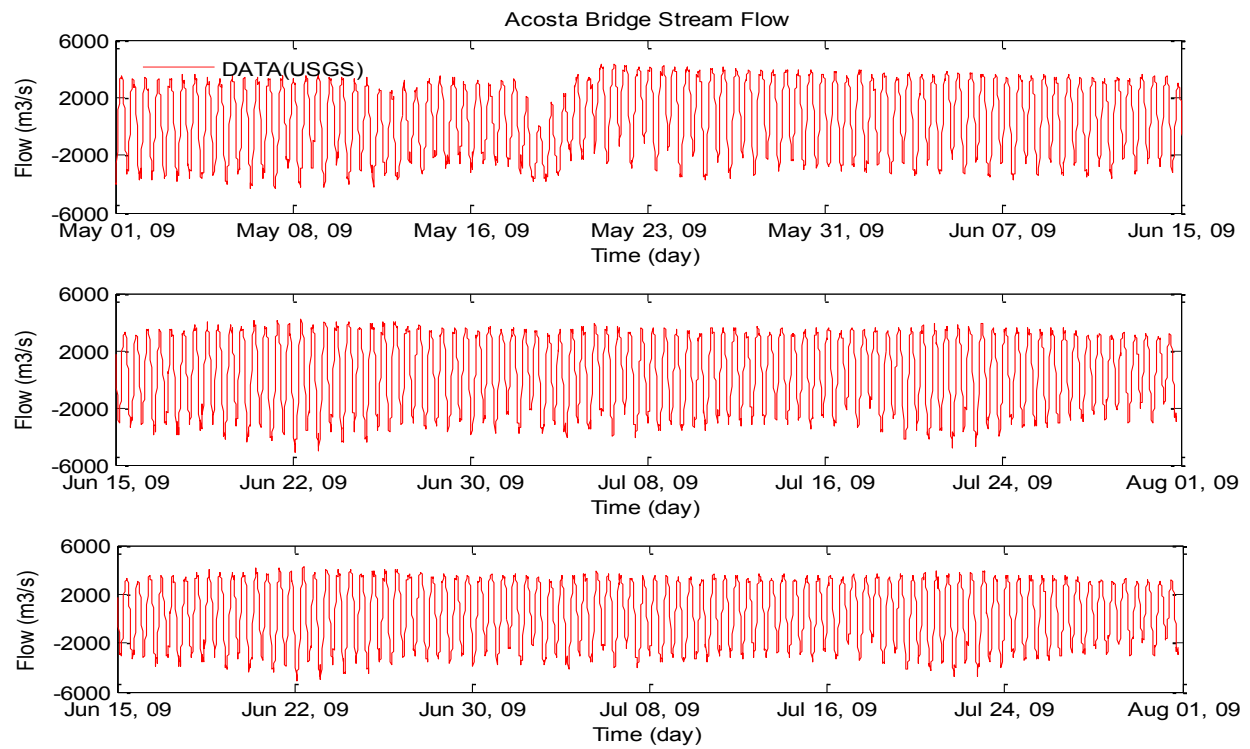


Figure 2.12. Flow discharge at Acosta Bridge.

### CHAPTER 3: MANNING'S ROUGHNESS COEFFICIENT

There are numerous approaches in determining the Manning roughness coefficient,  $n$ , to be used for open channel flow calculations and hydrodynamic modeling process for water flow in a natural channel (Arcement, 1989). Most methods use a description of the river or stream channel and its surfaces. Tables are available that give maximum, minimum and average Manning roughness ( $n$ ) values for different channel descriptions (Chow, 1959). There is also an approach that uses pictures of many natural open channels with known  $n$  values that can be matched to the channel of interest (Barnes, 1967). However, confident selection of values of the Manning roughness coefficient,  $n$ , usually requires considerable experience. Tables of computed  $n$  values for various channel conditions, illustrations, and stereoscopic color slides of channels for which  $n$  has been verified, are available to select an appropriate  $n$  value (Cowan, 1956). In hydrodynamic modeling, there is a need to reduce the uncertainty associated with identification of accurate Manning's  $n$ .

In river channels, the shape and area of the cross section of the flow can change along the stream, leading to non-uniform flows. On the other hand, flows that do not change over space are said to be uniform. Steady uniform flow in straight channels is a simplification of flows in the natural world, e.g., in rivers and in the ocean, although it reveals many fundamental aspects of those more complicated flows. From a physics standpoint, uniform flow occurs in a control volume when the frictional force is equal to the gravitational force. In this section, Newton's second law is applied to steady and uniform flow down an inclined plane. Because the flow is assumed to be steady and uniform, all of the forces in the streamline direction that are exerted upon the fluid within the free body at any given time must add up to be zero (Figure 3.1).



Writing Newton's second law for the balance of forces on the control flow equating the downslope driving force, caused by the downslope component of the weight of the fluid in the free body, with the resistance force exerted by the bottom boundary on the lower surface. This is balanced by the frictional force ( $\tau_b PL$ ) exerted by the bottom boundary where  $\tau_b$  is bottom shear stress,  $P$  is the wetting perimeter and  $L$  is the length of the control flow. There are also pressure forces acting parallel to the flow direction on the upstream and downstream faces of the free body, but because by assumption of uniformity the vertical distribution of these pressure forces is the same at every cross section, they balance each other out and cause no net force on the free body.

Graphic redacted, paper copy available  
upon request to home institution.

*Figure 3.1.* The control volume from (Chow, 1959).

#### Chezy and Manning Friction Formulations

In general, the mean velocity of uniform flow is described by the following formula:

$$V = CR^x S^y \dots\dots\dots(3-1)$$

in which  $C$  = friction coefficient,  $R$  = hydraulic radius (m),  $S$  = the channel slope (m/m) and  $x$  and  $y$  are exponents of  $R$  and  $S$ , respectively. The exponents vary with type of roughness

(laminar, turbulent, transitional, or mixed laminar-turbulent) and cross-sectional shape (arbitrary, hydraulically wide, rectangular, trapezoidal, triangular, or inherently stable).

In practice, there are two established uniform flow formulas: (1) the Chézy formula, and (2) the Manning formula. Variations of these formulas, the dimensionless Chézy and the Manning-Strickler are in current use.

The shear stress  $\tau_b$  along the channel bottom is modeled as a quadratic friction law:

$$\tau_b = \rho f V^2 \dots\dots\dots(3-2)$$

in which  $\rho$  = mass density,  $f$  = a type of friction factor (drag coefficient), and  $V$  = mean velocity.

The shear force developed along the wetted perimeter of a control volume of length  $L$  is:

$$F_s = \tau_b PL = \rho f V^2 PL \dots\dots\dots(3-3)$$

The weight of the water in the control volume is  $W$ . This gravitational force is resolved along the direction of motion to give:

$$F_g = W \sin \theta \dots\dots\dots(3-4)$$

Assuming a channel of small slope ( $\sin \theta \cong \tan \theta = S$ ) and equating Eq. 3-3 and 3-4 and reduce it to:

$$f V^2 = g \left( \frac{A}{P} \right) S = g R S \dots\dots\dots(3-5)$$

where  $R$  = hydraulic radius. Solving for  $V$ :

$$V = (g/f)^{1/2} (RS)^{1/2} \dots\dots\dots(3-6)$$

$$V = C (RS)^{1/2} \dots\dots\dots(3-7)$$

in which  $C$  = Chézy coefficient, defined as follows:

$$C = (g/f)^{1/2} \dots\dots\dots(3-8)$$

Equation 3-8 is called Chézy formula and a variation of the Chézy formula may be derived by solving for bottom slope  $S$  from Eq. 3-5:

$$S = f \frac{V^2}{gR} \dots\dots\dots (3-9)$$

The Manning formula, in SI units, is:

$$V = \frac{1}{n} R^{\frac{2}{3}} S^{\frac{1}{2}} \dots\dots\dots (3-10)$$

where  $n$  = Manning's friction coefficient, friction factor, or simply Manning's  $n$ , having units of  $\text{s/m}^{1/3}$ . In U.S. Customary units, the Manning formula is:

$$V = \frac{1.486}{n} R^{\frac{2}{3}} S^{\frac{1}{2}} \dots\dots\dots (3-11)$$

The factor 1.486 is required to express the original Manning equation (Eq. 3-10) in U.S. Customary units. In natural channels, the value of  $n$  may vary with stage and flow depth. This is attributed to variations in channel roughness with increasing stage, the effect of overbank flows, or morphological changes in total bottom friction as the flow rises from low stage to high stage (Simons and Richardson, 1966). Comparing Eqs. 3-7 and 3-11, the relation between Manning and Chézy and Manning's coefficients is obtained:

$$C = \frac{1}{n} R^{1/6} \dots\dots\dots (3-12)$$

Several correlations between Manning's  $n$  and particle size (grain diameter) have been developed. Williamson (1951) correlated the Darcy-Weisbach friction factor  $f$  with relative roughness to yield the following relation (Henderson, 1966):

$$f = 0.113 \left( \frac{k_s}{R} \right) \dots\dots\dots (3-13)$$

in which  $k_s$  = grain roughness, in length units, and  $R$  = hydraulic radius, in length units. Since  $f = 8 (g / C^2)$ , Eq. 3-13 reduces to:

$$C = \left( \frac{8g}{0.113} \right)^{1/2} \left( \frac{R}{k_s} \right)^{1/6} \dots\dots\dots(3-14)$$

In U.S. Customary units, Eq. 3-14 may be conveniently reduced to:

$$C = \frac{1.486R^{1/6}}{0.0311k_s^{1/6}} \dots\dots\dots(3-15)$$

Comparing Eq. 3-15 with Eq. 3-12,  $n$  can be expressed in terms of boundary roughness as follows ( $k_s$  in ft):

$$n = 0.0311k_s^{1/6} \dots\dots\dots(3-16)$$

A general expression for Manning's  $n$  in terms of relative roughness and absolute roughness is (Chow, 1959):

$$n = \left[ f \left( \frac{R}{k_s} \right) \right] k_s^{1/6} \dots\dots\dots(3-17)$$

which implies that in Eq. 3-16 the relative roughness is a constant (0.0311). Assuming that boundary roughness may be represented by the  $d_{84}$  particle size, i.e., that for which 84% of the grains (by weight) are finer, Eq. 3-16 converts to:

$$n = 0.0311d_{84}^{1/6} \dots\dots\dots(3-18)$$

Strickler used a constant (0.0342) for the function of relative roughness  $f(R/k_s)$ , and the median particle size  $d_{50}$  as the representative grain diameter, to yield:

$$n = 0.0342d_{50}^{1/6} \dots\dots\dots(3-19)$$

Since  $d_{84} > d_{50}$ , it is seen that the Strickler and Williamson equations are mutually consistent.

In streams with relatively stable boundaries, the total resistance to flow results from the interaction of many ingredients. Among them are particle size of streambed material, bank irregularity, vegetation, channel alignment, bed configuration, channel obstructions, converging or diverging streamlines, sediment characteristics, and surface waves. With the present

knowledge the quantitative effect of most of these factors is not determinable and must be estimated qualitatively. There is no exact method or procedure to estimate Manning's  $n$ . A proven set of recommendations is given below, as pertaining to the estimation of Manning's  $n$ :

- To consider the factors affecting the value of Manning's  $n$  and proceed accordingly.
- To consult a table of typical values, and to base the estimation on experience.
- To consult several pictorial collections for which the value of Manning's  $n$  has been documented with sufficient accuracy.
- To become acquainted with the appearance of typical channels for which the Manning's  $n$  values are known.

Chow (1959) presented a pictorial collection of 24 typical channels for which the Manning's  $n$  has been established. The values documented by Chow range from  $n = 0.012$  (a canal lined with concrete slabs, with a very smooth surface) to  $n = 0.150$  (a natural river in sand clay soil, irregular sides slopes and uneven bottom). Chow (1959) listed values of Manning's coefficient as low as  $n = 0.008$  (lucite, acrylic plastic) to as high as  $n = 0.200$ . These values are applicable to channel flow in the turbulent regime.

#### Factors Affecting Manning's $n$

Values of Manning's  $n$  are wide ranging. In natural stream channels, Manning's  $n$  can range from slightly lower than 0.020 for very large rivers featuring a relatively smooth boundary to higher than 0.200 for small creeks in steep mountain streams. The various factors affecting Manning's roughness coefficient are listed in Table 3.1.

Cowan (1956) developed a methodology for estimating Manning's  $n$  based on the cumulative effect of the various factors affecting bottom roughness. Cowan's equation is:

$$n = (n_0 + n_1 + n_2 + n_3 + n_4) m_5 \dots\dots\dots(3-20)$$

where  $n_0$  = basic  $n$  value for a straight, uniform, smooth channel,  $n_1$  = addition to account for surface irregularities,  $n_2$  = addition to account for variations in the size and shape of the cross section,  $n_3$  = addition to account for obstructions,  $n_4$  = addition to account for the effect of vegetation on flow conditions and  $m_5$  = factor to account for channel sinuosity (meandering).

Table 3.1

*Factors Affecting Manning's Roughness Coefficient*

Factor	How it affects
Surface roughness	Fine grain sizes lead to low values, while coarse grain sizes lead to high values.
Vegetation	Type, height, density, and spatial distribution of vegetation have a definite role in affecting flow velocity. Values of $n$ in vegetated channels may exceed 0.250, and in some cases, rise to 0.400 or greater.
Channel irregularities	Sand bars, ridges and depressions, and holes/humps in the channel bed create additional roughness in the form of local energy losses.
Channel alignment	Generally, a straight channel will feature a lower $n$ , while a sinuous channel will have a larger $n$ . Sinuosity may increase channel roughness by as much as 30% (Chow, 1959).
Aggradation and degradation	Changes in channel morphology will increase/decrease roughness in unpredictable ways. The effect will depend on the type of material forming the bed, the width-to-depth ratio (aspect ratio), and the quantity of sediment being transported (sediment load).
Channel obstructions	Log jams, bridge piers, and other obstructions tend to increase channel roughness. The effect will depend on the type of obstructions, their relative size, shape, number, and spatial distribution.
Size and shape of the channel	Generally, smaller channels have larger roughness, while larger channels have smaller roughness. The typically higher aspect ratio of larger channels tends to decrease roughness.
Stage and discharge	Roughness varies with stage and discharge in largely unpredictable ways. Mean velocities vary from very low stage to very high stage in complex patterns.
Season of the year	For vegetated channels, or channels lined with vegetation, surface roughness increases during the growing season, and decreases during the dormant season, subject to a latitudinal effect.
Suspended load and bedload	Sediment transport, either as suspended load or bed load, will consume additional energy and lead to increases in overall channel friction.

### Directional Manning's $n$

In a coastal river, the direction of the flow changes with the tide (Bertin, Fortunato, and Oliveira, 2009; Dodet et al., 2013; Malhadas, Leitao, Silva, and Neves, 2009; Olabarrieta, Warner and Kumar, 2011). Tidal flows in a coastal river are caused by the water level difference between the ocean and the estuary (Figure 3.2). Historically, the mathematical treatment of water wave theory by various investigators has been carried out with the assumption of a rigid, impermeable horizontal sea bed. In nature, the actual bottom stress varies drastically, e.g., from muds, which behave as viscous fluids, to rippled porous sand beds, to rocky bottoms. The degree of bed rigidity, the porosity, and the roughness all influence the water waves to varying degrees, as considered in the correction factors to Manning's  $n$  (Table 3.2).

Table 3.2

*Appropriate values to be used in Eq. 3-20*

Corrections to Manning's n			
Type of Material on Channel Boundary			Values
Earth	Sand, silt and clay boundary	$n_o$	0.02
Rock cut	Rock outcrop or rock boundary		0.025
Fine gravel	Gravel up to 8 mm diameter		0.024
Coarse gravel	Gravel of more than 8 mm diameter		0.028
Irregularity Modifier			
Smooth	Best regular condition	$n_1$	0
Minor	Good dredged channels, slightly eroded side slopes		0.005
Moderate	Fair to poor dredged channels, moderately eroded slopes		0.01
Severe	Badly eroded channels and highly irregular or jagged surfaces of channels excavated in rock		0.02
Cross Section Modifier			
Gradual	Smooth, or small variations	$n_2$	0
Alternating occasionally	Large and small sections alternate occasionally, occasional shifting of main flow from side to side		0.005
Alternating frequently	Large and small sections alternate frequently, frequent shifting of main flow from side to side		0.010-0.015
Obstructions Modifier			
Negligible	(a) The extent to which the obstructions occupy or reduce the flow area, (b) the character of the obstructions (sharp-edged or angular objects induce greater turbulence than curved, smooth-surface objects), and (c) the positioning and spacing of the obstructions, transversally and longitudinally, in the channel reach under consideration	$n_3$	0
Minor			0.010-0.015
Appreciable			0.015-0.030
Severe			0.030-0.060
Vegetation Modifier			
Low	Turf grasses or weeds, where the flow depth is 2 to 3 times the height of the vegetation	$n_4$	0.005-0.010
Medium	Turf grasses or weeds, where the flow depth is 1 to 2 times the height of the vegetation		0.010-0.025
High	Turf grasses or weeds, where the flow depth is about equal to the height of the vegetation		0.025-0.050
Very high	Turf grasses or weeds, where the flow depth is less than one-half (1/2) the height of the vegetation		0.050-0.100
Channel sinuosity or Meandering Modifier			
Low	Sinuosity less than 1.2	$m_5$	1
Medium	Sinuosity between 1.2 and 1.5		1.15
High	Sinuosity greater than 1.5		1.3



In the 16-mile stretch of the LSJR where the velocity measurements were taken, the currents exhibit mostly progressive wave characteristics, meaning that the maximum strengths of flood and ebb occur near the times of high and low water, respectively. This relationship varies along the river, depending on the distance from the mouth of the river, the water depth, and other physical factors. At the river entrance, the maximum flood and ebb currents occur approximately one hour before the high and low tides at the river entrance (NOAA, 1999).



Figure 3.2. Ebb and flood tide flow.

The horizontal velocity of a progressive water wave is given by (Dean and Dalrymple, 1991):

$$u = \frac{g a k}{\sigma} \frac{\cosh k(h+z)}{\cosh kh} \cos(kx - \sigma t) \dots\dots\dots(3-21)$$

where,  $\sigma (=2\pi/T)$  is angular frequency,  $T$  is wave period,  $a$  is the amplitude of the wave,  $k(=2\pi/L)$  is the wave number,  $L$  is the wave length and  $h$  is water depth. The presence of waves over the bed causes significant bed deformation and stresses (Dean and Dalrymple, 1991). Little has been done to investigate if the Manning's  $n$  changes with time and direction of the flow due to the additional wave-induced bottom stress. Dean and Dalrymple (1991) obtained the instantaneous shear stress exerted on the bed from the Newtonian shear stress term for waves over smooth, rigid impermeable bottoms in a laminar boundary layer:

$$\tau_{xz} = \rho \sqrt{\frac{\nu}{\sigma}} \frac{g a k}{\cosh kh} \cos(kx - \sigma t - \pi/4) \dots\dots\dots(3-22)$$

where  $\rho$ = mass density,  $\nu(= \frac{\mu}{\rho})$  is the kinematic viscosity,  $\mu$  is dynamic viscosity. The bed shear stress is harmonic in time and lags the free surface displacement by  $45^\circ$ . During ebb tide, the flow is directed downstream to the ocean and wave induced shear stress is added to the shear stress caused by the weight of the water and pressure forces. This could cause significant bed deformation and stress which eventually changes the roughness of the bottom surface, whereas during the flood tide the wave induced shear stress reduces the shear stress caused by the weight of the water and pressure forces which can also change the bottom roughness. A thorough investigation needs to be done to know exactly how the flow direction affects the bottom roughness of the sea bed for the two scenarios.

A careful analysis is also needed on the base values for the Manning's  $n$  and whether the correction factors remain the same for different flow direction, i.e., when it is ebb and when it is flood. Values of the roughness coefficient,  $n$ , may be assigned with their corrections for conditions that exist at the time of a specific flow event, for average conditions over a range in stage, or for anticipated conditions at the time of a future event. Channel irregularities, alignment, obstructions, vegetation, and meandering increase the roughness of a channel. The value for  $n$  must be adjusted accordingly by adding or subtracting increments of roughness to the base value,  $n_0$ , for each condition that increases or decreases the roughness based on direction of the flow. The roughness coefficients apply to a longitudinal reach of channel and (or) flood plain. The evaluation of  $n$  is complicated by the depth of flow (Arcement and Schneider, 1989). If the depth of flow is shallow in relation to the size of the roughness elements, the  $n$  value can be appreciably large. The  $n$  value decreases with increasing depth, except where the channel banks are much rougher than the bed or where dense brush overhangs the low-water channel. Chow's assumption of base value of  $n_0$  is for a straight, uniform, smooth channel in natural materials. In reality, there is no straight, uniform and smooth natural channel.

Obstructions, such as logs, stumps, boulders, debris, pilings, and bridge piers disturb the flow pattern in the channel and increase apparent roughness. The amount of increase depends on the shape of the obstruction; the size of the obstruction in relation to that of the cross section; and the number, arrangement, and spacing of obstructions. For bottom stress of two-dimensional flows, one considers both bottom and lateral obstructions to affect the bottom roughness. The slope of channel becomes a source of obstruction during flood tide where the flow is directed upstream. The effect of obstructions on the roughness coefficient is a function of the flow

velocity. When the flow velocity is high, an obstruction exerts a sphere of influence that is much larger than the obstruction itself because the obstruction affects the flow pattern for considerable distances on each side (Arcement and Schneider, 2009).

In general, Eq. 3-20 contains different correction factors which vary in time and with the flow direction. This requires a more accurate parameterization of Manning's  $n$  in the hydrodynamic model, such that the formulation considers time and flow direction. Although the hydrodynamic is capable to parameterize Manning's  $n$  on a spatial basis, to date, Manning's  $n$  is treated as an isotropic scalar, i.e., independent of the flow direction (Atkinson et al., 2015). The purpose of the next chapter (Chapter 4) is to observe directional characteristics of the Manning's  $n$  (anisotropic) based on observation data to include the directional behavior to the modeling in order to increase the accuracy of ADCIRC hydrodynamic modeling.

## CHAPTER 4: HYDRODYNAMIC MODELING

The ADCIRC (ADvanced CIRCulation) hydrodynamic model is a continuous-Galerkin, finite-element based model for coastal oceans, inlets, rivers and floodplains (Luettich and Westerink, 2006). ADCIRC solves the shallow water equations in the Reynolds averaged form by depth-integrating the more general Navier-Stokes equations (conservation of mass and conservation of momentum). For shallow water flows, the horizontal scale is much greater than the vertical scale, such that it is safe to neglect vertical accelerations as well as vertical variation of the flow velocity. ADCIRC solves the shallow water equations (Kinnmark, 1985) in the form of the generalized wave continuity equation (GWCE) (Lynch and Gray, 1979; Kolar et al., 1994). A continuous-Galerkin finite element scheme is applied over linear triangles in space and a three-level implicit scheme is used to advance the solution forward in time (Westerink et al., 2008).

### Shallow Water Equations

The shallow water equations include two conservation equations; conservation of mass and momentum equations which describe fluids motion. The equation for conservation of mass entails that mass be balanced by the divergence of flux (Nathan, 2007):

$$\frac{\partial h}{\partial t} + \frac{\partial}{\partial x}(hu) + \frac{\partial}{\partial y}(hv) = 0 \dots\dots\dots(4-1)$$

where  $u, v$  are the depth averaged mean velocities in the Cartesian  $x$  and  $y$  directions respectively,  $h$  is the depth of the water column from free water surface to bottom, and  $t$  is time. The conservation of momentum entails the balance of local accelerations with advective accelerations and pressure, viscous and frictional effects:

$$h \frac{\partial u}{\partial t} + h(u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y}) - \frac{h}{\rho} E \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + gh \left( \frac{dz}{dx} + \frac{dh}{dx} \right) = T_{sx} - T_{bx} - C_x \dots\dots\dots(4-2)$$

$$h \frac{\partial v}{\partial t} + h(u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y}) - \frac{h}{\rho} E \left( \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) + gh \left( \frac{dz}{dy} + \frac{dh}{dy} \right) = T_{sy} - T_{by} - C_y \dots\dots\dots(4-3)$$

where  $\rho$  is water density,  $g$  is gravitational acceleration,  $E$  is the eddy viscosity, and  $z$  is the bottom elevation.  $T_{bx}$  and  $T_{by}$  are the bottom stress terms.  $T_{sx,sy}$  and  $C_{x,y}$  are terms representing surface wind stress and Coriolis forcing, respectively.

ADCIRC uses a generalized slip formulation for the bottom stress terms in the governing equations (Luettich and Westerink, 2004). The bottom stress is calculated by:

$$\frac{\vec{\tau}_b}{\rho_o} = K_{slip}(\vec{u} + \vec{v}) \dots\dots\dots(4-4)$$

where  $\rho_o$  is the reference density of water,  $\vec{u}, \vec{v}$ , are the  $x, y$  components of depth averaged velocity, and  $K_{slip}$  may be a constant giving a linear slip boundary condition, or it may be calculated as,

$$K_{slip} = C_d \sqrt{u^2 + v^2} \dots\dots\dots(4-5)$$

resulting in a quadratic slip boundary condition on the bottom with the necessary specification of a quadratic drag coefficient  $C_d$ . In this case, the bottom stress terms in the momentum equations become:

$$\frac{\vec{\tau}_b}{\rho_o} = C_d U(\vec{u} + \vec{v}) \dots\dots\dots(4-6)$$

where  $U = \sqrt{u^2 + v^2}$  is the velocity magnitude and  $C_d = gn^2/h^{1/3}$  (Luettich and Westerink, 2006a) where  $g$  is gravitational acceleration;  $h$  is the water depth; and  $n$  is the Manning's roughness coefficient. An equivalent Manning's  $n$  value to the drag coefficient can be calculated using a depth scale from the following equation:

$$n = \frac{h^{1/6} \sqrt{C_d}}{g^{1/2}} \dots\dots\dots(4-7)$$

### Bottom Friction Formulations in ADCIRC

The ADCIRC model is a computational code that is compiled to simulate flow processes based on input files that describe the region of interest and its characteristics, including boundary conditions and forcing mechanisms. The critical inputs for this study are the computational mesh, surface characteristics file, and meteorological forcing files, such as wind and pressure fields. The surface characteristic of focus in this study is the spatial/nodal attribute for bottom friction. Currently, ADCIRC can employ Constant Linear, Constant Quadratic, Constant Hybrid, Varying Linear and Varying Quadratic friction formulations to parameterize bottom friction. The "Constant Linear" option should only be used for analytical cases when verifying the code, such that linear friction neglects nonlinearity and causes over-damping in deep water. For "Constant Quadratic," nonlinearity is accounted for and the value of  $\text{TAU0}$  should be set based on the principle depth, where  $\text{TAU0}$  is a weighting factor in the GWCE that weights the relative contribution of the primitive and wave portions of the GWCE (Kolar et al., 1994). A value of  $\text{TAU0} = 0.005$  is suggested for deeper water (greater than 10 m depth). For shallow water,  $\text{TAU0} = 0.02$  is recommended. If the domain includes both shallow and deep water, the user may want to consider the Varying Quadratic option. ADCIRC also uses the hybrid formulation to vary bottom friction automatically with depth, i.e. in deep water, the coefficient of friction does not change and a Quadratic friction formulation results and in shallow water the coefficient of friction increases as the depth decreases. Simulations were run by using the three available options on ADCIRC, Manning's  $n$ , Quadratic and Hybrid friction formulations. Note that these friction formulations are able to be parameterized within the model on a spatially variant basis,

but they are not able to be parameterized on a directionally variant basis, like with the redeveloped friction formulation presented later in this thesis.

### ADCIRC Model Setup

The source code for the ADCIRC model, along with the user's manual and other resources are available online at <http://www.adcirc.org>. The development of ADCIRC is generally attributed to Luetich & Westerink (1991). Since that time, many modifications and upgrades to the model have been made and ADCIRC presently enjoys very wide use among the academic community and federal agencies such as the Army Corps of Engineers, NOAA, and the Naval Research Laboratory. In this thesis, ADCIRC was used in barotropic mode (i.e., pressure-based and neglecting any density-driven effects), where the model solves for water surface elevation and depth-integrated velocity. ADCIRC is a Fortran program which requires, at a minimum, two input files to run, fort.14 (mesh) and fort.15 (mesh parameter). In addition to these files, there are optional files that may or may not have to be present, depending upon which features the user wishes to incorporate into a particular run. In this study, the input files listed in Table 4.1 were used to run the model.



Table 4.1

*Model Input Files (Simulations 1, 2 and 3)*

Input files	File description		
fort.14	Triangulated mesh with 284,739 number of elements and 147,688 nodes.		
	Simulation 1 (SIM1)	Simulation 2 (SIM2)	Simulation 3 (SIM3)
	TIME STEP = 0.50 SECONDS	TIME STEP = 0.50 SECONDS	TIME STEP = 0.50 SECONDS
	COLD START	COLD START	COLD START
fort.15	LENGTH OF SIMULATION=92 DAYS	LENGTH OF SIMULATION=92 DAYS	LENGTH OF SIMULATION=92 DAYS
	NOLIBF = 1	NOLIBF = 1	NOLIBF = 2, HBREAK=20, FTHETA=10, FGAMMA=0.3333
fort.13	mannings_n_at_sea_floor, n=0.02 ( constant)	quadratic bottom friction FFACTORMIN=0.0025	hybrid bottom friction FFACTORMIN=0.0025
fort.19	Tidal elevation forcings (full tidal forcing), along the 60° West Meridian		
fort.20	Fresh water river inflows (at 1800-sec time intervals) are applied at 21 tributary locations		
fort.22	Meteorological inputs (winds and pressures) are updated where these forcing data are applied over the surface of the model domain.		

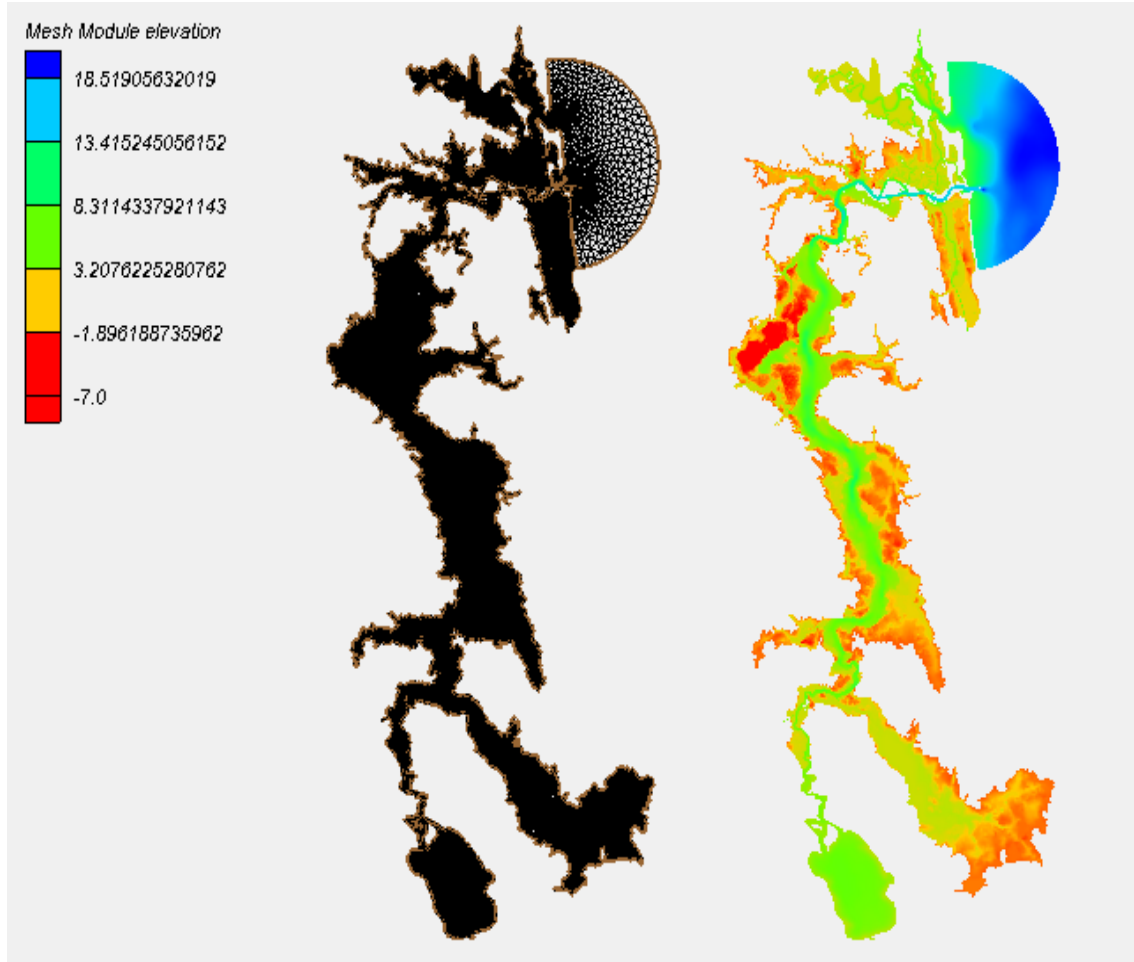
*Fort.14 File – Domain Mesh*

The foundation of an ADCIRC simulation is the model domain mesh. For the simulations performed in this study, a large domain is utilized that covers the Western Atlantic Ocean, Caribbean Sea and Gulf of Mexico waters to the west of the 60°W longitude (Figure 4.1). The large domain ensures that tide- and wind-driven hydrodynamics are accurately represented and delivered to the local domain of LSJR in the form of boundary conditions.

Graphic redacted, paper copy available  
upon request to home institution.

*Figure 4.1.* Finite element mesh for the Western North Atlantic Tidal model domain.

The finite element mesh (FEM) model developed by Bacopoulos et al. (2012) was mapped out and cutoff based on the number of inflow locations for the hydrodynamic modeling process. After defining the mesh size distribution the other key ingredient, bathymetry, or depth originated from (Bacopoulos et al., 2009) is applied. Figure 4.2 shows the triangular mesh that is used for this study which consists of 147,688 computational nodes. Note the unstructured triangulation and that it is able to efficiently represent depth variations and shoreline configuration of the river. The boundary conditions included inflow at the upstream boundary and tidal fluctuations along the open-ocean boundary, as supplied by the WNAT model domain (Figure 4.1).



*Figure 4.2* ADCIRC mesh triangulation (left) with colored elevation (right). At this scale, the nodal density makes the triangulation difficult to view.

#### Fort.15 File – Parameter File

Once a good model mesh is completed, the fort.14 file becomes essentially static, such that it does not need to be altered from simulation to simulation. The fort.15 file, on the other hand, contains many simulation-specific parameters that may need to be adjusted with any change in the attribute file (fort.13). For a complete description of the fort.15 file, the reader is referred to the ADCIRC documentation provided at <http://www.adcirc.org>. Only the relevant items of the fort.15 file will be covered in this section. The fort.15 file contains many parameters

that govern how ADCIRC will be run. If it is to be run in two-dimensional or three-dimensional mode, coordinates must be specified or coordinates must be projected (x, y). There are also some important parameters that control the manner in which nonlinearity is dealt. Finite-amplitude effect was considered by setting NOLIFA to be nonzero. In addition to simply considering finite-amplitude effects, the ADCIRC model has the distinct advantage of being able to model wetting and drying effects. In other words, as the tide ebbs, some elements will become ‘dry’ as the water contact line moves away from shore. This feature of ADCIRC is turned on by setting NOLIFA to have a value of 2. The model time step was set  $DT=0.5$  s for Simulation 1 (SIM 1) for initial run. The middle section of the fort.15 file contains information about how the model is to be forced (e.g., with tides, winds and inflows). The last major portion of the fort.15 file is to specify the required output data. All the ADCP measurement locations for the velocity data and validation data stations were listed here.

#### Fort.13 File – Nodal Attribute File

Nodal attributes that are constant in time but spatially variable are defined in this file. The basic file structure can be obtained from <http://www.adcirc.org>. Presently, ADCIRC does not have the capability to vary nodal attributes over time. To date, simulations focus on running with different spatial parameterizations of bottom roughness. The Manning’s n coefficient is typically specified as a variable spatial parameter dependent on the surface characteristics of the seabed. The nodal assignment of bottom friction attribute is usually based on data (e.g. satellite images used to define land classifications) that may be incomplete or contain data from different, inapplicable time periods (Medeiros et al., 2012). Furthermore, data is often collected on a sub-grid scale and an upscaling procedure (e.g. local spatial averaging) must be used. As a result,

even when applied spatially, the Manning's  $n$  coefficients used in coastal ocean models often contain large amounts of uncertainty. The uncertainty in the Manning's  $n$  coefficient on any given domain leads to uncertainties in common quantities of interest computed from the solution of a coastal ocean model (e.g. maximum water elevations and currents). Moreover, the solution of the model and thus the quantities of interest are often highly sensitive to changes in these parameters (Mayo, Butler, Dawsom, and Hoteit, 2014).

The first three simulations, i.e., those using the standard ADCIRC friction formulations (see Table 4.1), were run successfully and the results were compared to the ADCP measurements (Figure 4.3 and 4.4). The flood currents are approximately 0.6 m/s while the ebb currents are approximately 0.75 m/s. The apparent decay in tidal current with increasing duration of the deployment is because of the boat transect going upstream in the LSJR. The model is able to reproduce the greater ebb currents relative to lesser flood currents and the tidal decay with distance upstream. All the three simulations seems to fairly capture the velocity magnitude on the flood tide on both dates (deployments), whereas all the simulations over predicted in the ebb direction. Table 4.2 shows the ratios of average velocity magnitudes for both dates (deployments) to compare model-data disparity. Simulation 3 was used to calculate the average velocity magnitudes for the model. The model simulated 1.53 times larger value of the average velocity magnitude during ebb flow whereas it simulated approximately the same value of average speed during flood flow showing the model-data disparity for ebb and flood current.

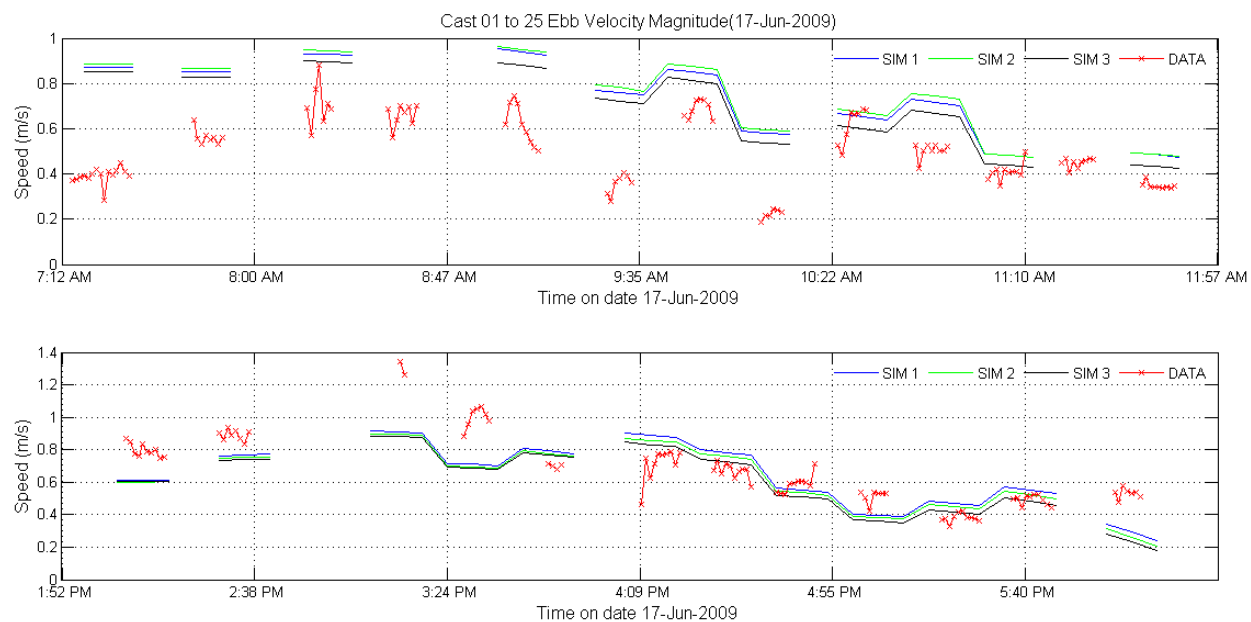


Figure 4.3. Simulations 1, 2 and 3, 17-Jun-2009.

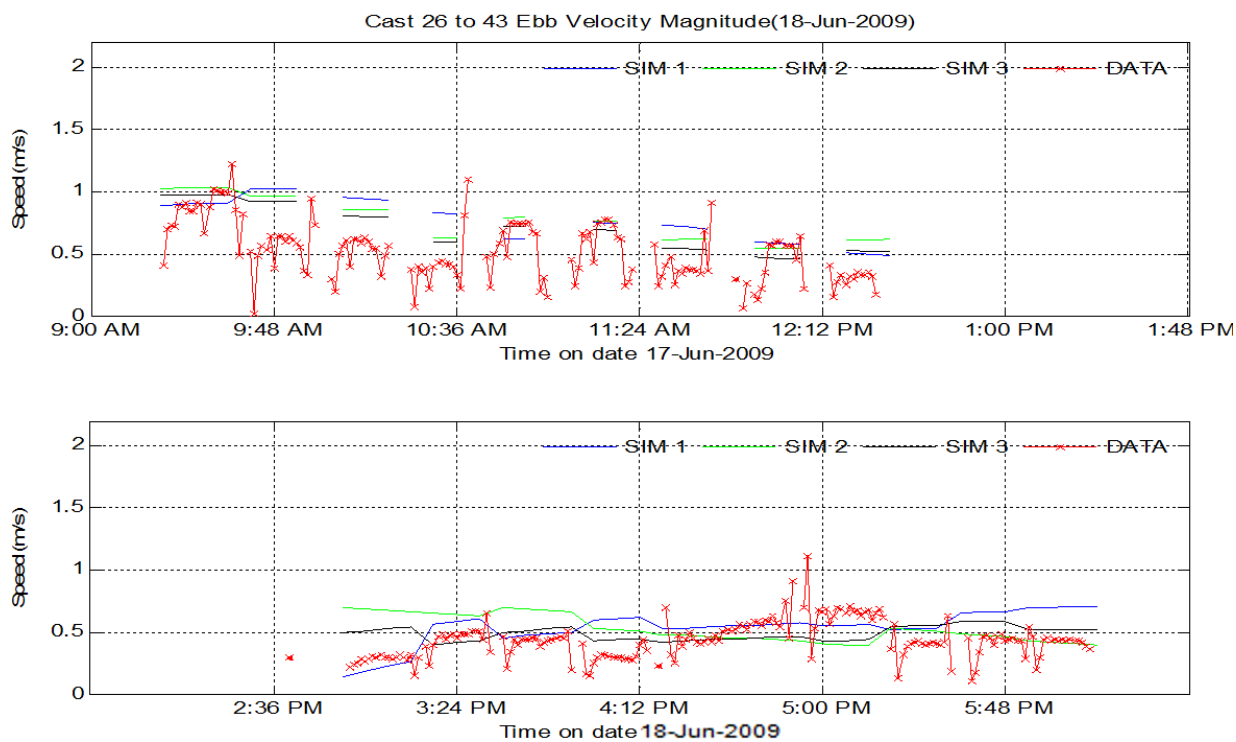


Figure 4.4. Simulations 1, 2 and 3, 18-Jun-2009.

### Development of Directional Manning's n

The major outcome of this thesis is utilization of a directionally variant Manning's n formulation, based on assimilated data, for improved parameterization of bottom roughness in the ADCIRC model. Table 4.3 shows the scenarios run in the model. The fourth and fifth simulations consider Manning's n to be variable with the direction of the flow and thusly with time. The numerical code of ADCIRC was modified to incorporate directional variability of bottom roughness based on the direction of the time-dependent velocity field. The crux of the directionally variant bottom roughness formulation is with the assimilation of the velocity dataset (see Chapter 2) for model calibration and parameter estimation.

From the quadratic bottom friction formulation, bottom stress can be computed using Eq. 4-5 and substituting  $C_d = gn^2/h^{1/3}$  and rearranging for  $\tau_b$ :

$$\vec{\tau}_b = (g\rho_o n^2/h^{1/3})U(\vec{u} + \vec{v}) \dots\dots\dots(4-7)$$

where  $U = \sqrt{u^2 + v^2}$  is the velocity magnitude. A relationship between the Manning's n used for the model and the "true" Manning's n can be obtained by equating the bottom stress magnitude for the model with that of the observed data. It is assumed that acceleration due to gravity, the bathymetric depth and the density remains the same for the model and observed data, which implies  $\tau_b(model) = \tau_b(observed)$ .

$$g\rho_o n_{model}^2/h^{1/3}U_{model} = g\rho_o n_{data}^2/h^{1/3}U_{data} \dots\dots\dots(4-8)$$

$$n_{data} = n_{model} \sqrt{\frac{U_{model}}{U_{data}}} \dots\dots\dots(4-9)$$

From this, it is shown that Manning's n varies with the flow direction and that  $n_o = \frac{n_{data}}{n_{model}} =$

$\sqrt{\frac{U_{model}}{U_{data}}}$  is the n-factor used to describe the ratio of "true" Manning's n to the one used for the

model. Using the data assimilation described above, bottom roughness factors were computed for the two deployments of ebb and flood velocity measurements (Table 4.2). As shown, the roughness factor  $n_0$  averages for the ebb and flood direction are 1.2107 and 1.024, respectively, which suggests that bottom roughness in the ebb flow direction ( $n_{\text{ebb}}$ ) is higher by 15.4% relative to the bottom roughness in the flood flow direction ( $n_{\text{flood}}$ ).

Table 4.2

*Computation of Roughness Factor  $n_0$* 

Day	Flow direction	Average- $U_{\text{data}}$ (m/s)	Average- $U_{\text{model}}$ (m/s)	$U_{\text{model}}/U_{\text{data}}$ (dimensionless)	Average- $n_0$ (dimensionless)
Both days 17-Jun-09 18-Jun-09	Ebb flow	0.4735	0.7382	1.5590	1.1904
	Flood flow	0.7144	0.6435	0.9007	0.9587
	Ebb flow	0.5098	0.7629	1.4965	1.2238
	Flood flow	0.4070	0.5157	1.2670	1.1220
	Ebb flow	0.4884	0.7493	1.5342	1.2107
	Flood flow	0.5914	0.5924	1.002	1.0240

In order to run the vector-based Manning's  $n$  parameterization, the source code of ADCIRC was modified. All substantive code changes are highlighted in yellow and are shown in Appendix A. The source code which needs to be modified is the subroutine entitled nodalattr.F. The nodalattr.F reads the fort.13 file and initializes time invariant attributes at each node. This subroutine manages nodal attribute data, including bottom friction,  $\tau_0$ , startdry, directional wind speed reduction, and etc. The concept of the directionally variant bottom friction



formulation is very similar to the surface directional wind speed reduction by how it treats the process directionally (Atkinson et al., 2015). A new attribute was created, named as `directional_effective_roughness_factor`, where this attribute is directional, such that there are twelve values, each representing the roughness as “seen” in twelve different compass directions at each node. The orientation of the 12 values follows the trigonometric convention, such that zero degrees represents ebb flow (where maximum Manning’s  $n$  is applied and hereafter is referred to as  $n_{ebb}$ ), then the values proceed in counter clockwise fashion. In other words, the first value at a node is applied to flow direction in the alignment of flow towards the ocean, the sixth value at 180 degrees applies the minimum Manning’s  $n$  which hereafter is referred to as  $n_{flood}$ , and it varies directionally when the flow direction is between ebb and flood directions.

The time stepping loop is used to inform the bottom roughness formulation about the flow direction for all mesh nodes. As described in Table 4.2, the  $n_0$  value is anisotropic. Figure 4.5 shows the basic modification to the `nodalattr.F` file by introducing a loop to calculate the direction of the flow and to assign a roughness factor  $n_0$  for a given node at every time step. For the case of hybrid friction formulation, ADCIRC uses the hybrid bottom friction relationship to vary  $C_d$  automatically with depth according to:

$$C_d = FMIN \left[ 1 + \left( \frac{HBREAK}{h} \right)^{FTHETA} \right]^{\frac{FGAMMA}{FTHETA}} \dots\dots\dots (4-10)$$

where  $FMIN$  is the minimum BFC;  $HBREAK$  is the break depth;  $FTHETA$  is a dimensionless parameter that controls how fast the BFC approaches its upper and lower limits;  $FGAMMA$  is a dimensionless parameter that controls how quickly the BFC increases as water depth decreases (Luettich and Westerink, 2006b).

The equivalent equation for Eq. 4.10 can be obtained for Manning's  $n$  by substituting Eq. 4.10 into equation 4-7 yielding  $n$  values as a function of depth.

$$n = \frac{h^{1/6} \sqrt{FMIN \left[ 1 + \left( \frac{HBREAK}{h} \right)^{FTHETA} \right]^{\frac{FGAMMA}{FTHETA}}}}{g^{0.5}} \dots\dots\dots (4-11)$$

To implement the assignment of  $n_0$  nodal values, the directions of the rectilinear flow in the LSJR need to be analyzed since the alignment of the river basin is not unidirectional towards the ocean (due to the meandering nature of the channel). Matlab code was used to generate the LSJR basin roughness factor based on the local orientation of the channel. Once the nodalattr.F file was modified and ready for compilation (Appendix A), all of the changes were saved into a different source code (src) directory on the UNF high-performance computing cluster OPUS. The compilation procedure consisted of: 1) typing “make clean” (without the quotes) at the command line for the metis and work folder; 2) still being inside the work folder, typing “make adcprep”, “make adcpst” and “make padcirc” and at the command line. The makefile is set up to recognize known architecture and compile combinations (<http://adcirc.org>). All went successfully and the three binary files adcprep, adcpst and padcirc were generated, which are the executables used to run the ADCIRC simulation.

Table 4.3

*Model Input Files (Simulations 4 and 5)*

Input files	File description	
fort.14	Triangulated mesh with 284,739 number of elements and 147,688 nodes.	
	Simulation 4	Simulation 5
	TIME STEP = 0.25 SECONDS	TIME STEP = 0.25 SECONDS
	COLD START	COLD START
fort.15	LENGTH OF SIMULATION=92 DAYS	LENGTH OF SIMULATION=92 DAYS
	NOLIBF = 1	NOLIBF = 2, HBREAK=20, FTHETA=10, FGAMMA=0.3333
fort.13	directional_effective_roughness_factor, in 12 directions mannings_n_at_sea_floor, n=0.02 ( constant)	directional_effective_roughness_factor, in 12 directions hybrid bottom friction, FFACTORMIN=0.0025
fort.19	Tidal elevation forcing (full tidal forcing), along the 60° West Meridian	
fort.20	Fresh water river inflows (at 1800-sec time intervals) are applied at 21 tributary locations	
fort.22	Meteorological inputs (winds and pressures) are updated where these forcing data are applied over the surface of the model domain	

```

REAL(SZ) n01    ! ManningsN factor for a particular node, for particular direction
REAL(SZ) angle  ! Flow direction
INTEGER iflowdir ! code for wind direction

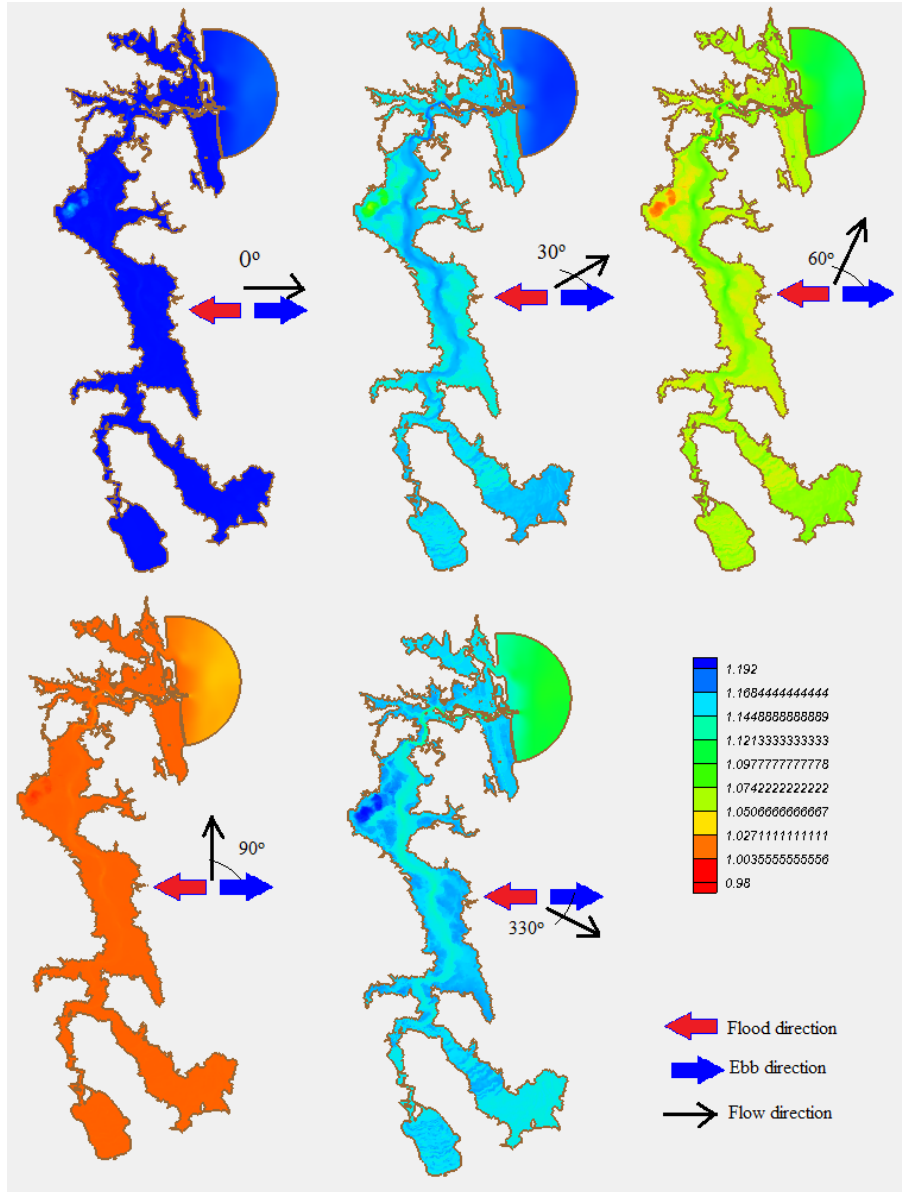
compute direction that the flow is coming from
if((UU1(I).eq.0).and.(VV1(I).eq.0))then
    angle=0.d0
else
    angle=atan2(VV1(I),UU1(I))
endif
angle=360.*angle/(2*3.141592654d0)
iflowdir=0
if((angle.gt.-15.).and.(angle.le.15)) iflowdir=1
if((angle.gt.15.).and.(angle.le.45)) iflowdir=2
if((angle.gt.45.).and.(angle.le.75)) iflowdir=3
if((angle.gt.75.).and.(angle.le.105)) iflowdir=4
if((angle.gt.105.).and.(angle.le.135)) iflowdir=5
if((angle.gt.135.).and.(angle.le.165)) iflowdir=6
if((angle.gt.165.).and.(angle.le.180)) iflowdir=7
if((angle.gt.-45.).and.(angle.le.-15)) iflowdir=12
if((angle.gt.-75.).and.(angle.le.-45)) iflowdir=11
if((angle.gt.-105.).and.(angle.le.-75)) iflowdir=10
if((angle.gt.-135.).and.(angle.le.-105)) iflowdir=9
if((angle.gt.-165.).and.(angle.le.-135)) iflowdir=8
if((angle.ge.-180.).and.(angle.le.-165)) iflowdir=7

define roughness from values
n01=z0land(NodeNumber,iflowdir)
Step 0. Convert Manning's N to Cd, if necessary.
IF (LoadManningsN) THEN
    DO I=1, NP
        FRIC(I)=g*n01*ManningsN(I)**2.d0
        /(( DP(I)+IFNLFA*ETA2(I) )**(1.d0/3.d0) ) ! sb46.28sb02
        !sb46.28sb02 Lower limit is applied here.
        IF(FRIC(I).LT.BFCdLLimit) THEN
            FRIC(I) = BFCdLLimit
        ENDIF
    ENDDO
ENDIF

Step 1. Apply friction arising from turbulent viscous interaction
with the sea floor.
DO I=1, NP
    UV1=SQRT(UU1(I)*UU1(I)+VV1(I)*VV1(I))
    H1=DP(I)+IFNLFA*ETA2(I)
    TK(I)= FRIC(I) *
    ( IFLINBF + ! linear
    (UV1/H1) * (IFNLBF ! nonlinear
    + IFHYBF*(n01**2)*(1+(HBREAK/H1)**FTHETA)*(FGAMMA/FTHETA)) ) ! hybrid
END DO

```

Figure 4.5. Modified subroutine (Apply2DBottomFriction). All code changes are shown in Appendix A.



*Figure 4.6.* Roughness factor contours for different sample flow directions. (Note: Ebb and flood direction are shown in arrows for reference only, they tend to adjust with the orientation of the river channel.)

#### Validation of the Modified ADCIRC Model

Validation of the modified ADCIRC model was conducted with the goal of validating the new code (Figure 4.5) which was written to simulate now with a vector-based directional bottom roughness. For validation, the new code was tested by using a roughness factor of  $n_0 = 1$  for the

12 directions, which is analogous to running with the original code (i.e., isotropic BFC). The result was found to match with the original simulations, implying the modified code is able to reproduce the same as the original code for the case of isotropic bottom roughness. After validation of the modified code, the model was run with spatially and temporally variable Manning's  $n$  based on flow direction for the entire mesh. Figures 4.7 and 4.8 show the expected result that an overall increase in effective Manning's  $n$  values in the ebb direction for SIM 4 and 5 causes a fairly uniform decrease in the velocity during ebb. This is more visible in SIM 5 when the bottom friction is increased at the shallow end of the depth range, which is due to the decrease in depth, thus resulting in a larger drag coefficient to be used in the hybrid friction formulation.

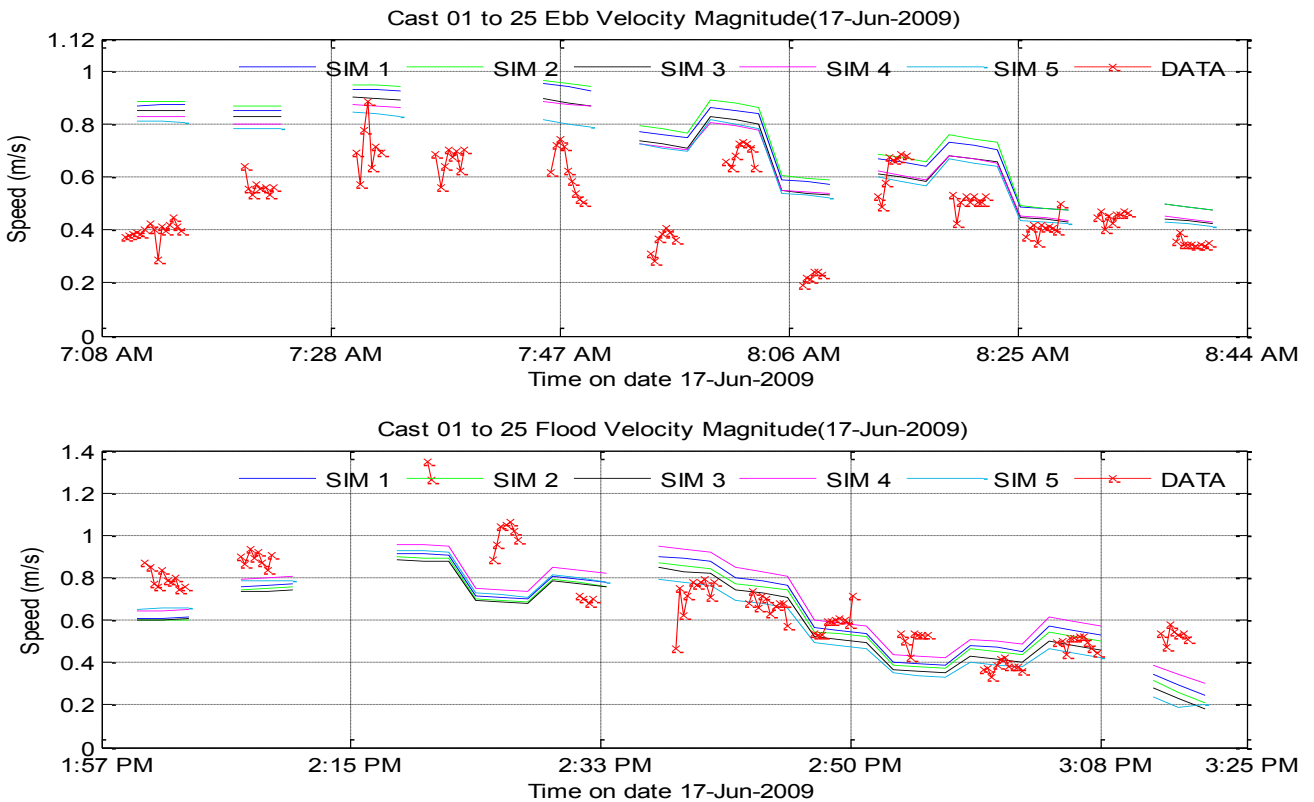


Figure 4.7. Simulations 1 to 5 (Locations 01 to 25).

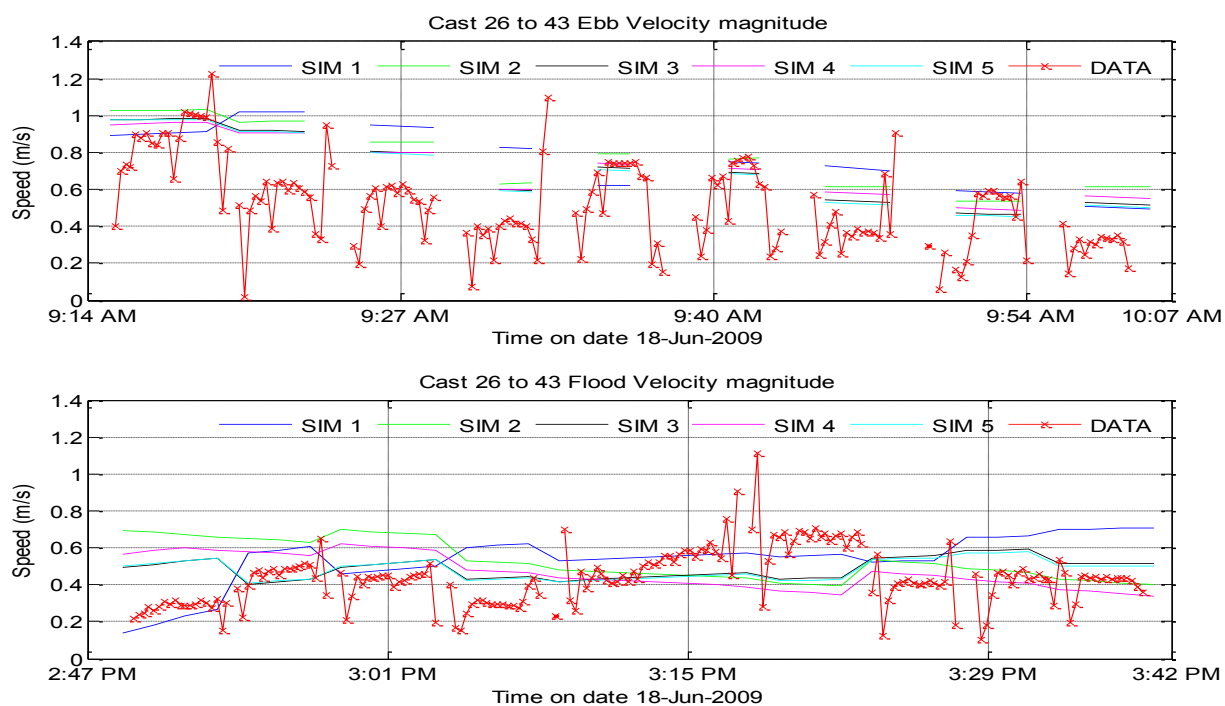


Figure 4.8. Simulations 1 to 5, location 26 to 43.

## CHAPTER 5: RESULTS AND DISCUSSION

The model-data disparity associated with the plots shown in Figures 4.7 and 4.8 was quantified using a root-mean-square error (RMSE) and coefficient of determination ( $R^2$ ). The RMSE is a frequently used measure of the differences between values (sample and population values) predicted by a model or an estimator and the values actually observed (Chai and Draxler, 2014). The RMSE has the same units as the measured and calculated data, such that smaller values indicate better agreement between measured and calculated values. RMSE is defined as follows:

$$RMSE = \sqrt{\frac{\sum (Obs_i - Sim_i)^2}{N}} \dots \dots \dots (5-1)$$

where  $Obs_i$  relates to the historical observation data at time  $i$ ,  $Sim_i$  corresponds to the model data at time  $i$ , and  $N$  refers to the total number of observations available for the error estimation. The coefficient of determination ( $R^2$ ) is a measure of how well the model represents the observation data and it is defined as the ratio of the explained variation to the total variation (Legates and McCabe, 1999).

$$R^2 = \left( \frac{N \sum Obs_i Sim_i - (\sum Obs_i)(\sum Sim_i)}{\sqrt{N(\sum Obs_i^2) - (\sum Obs_i)^2} \sqrt{N(\sum Sim_i^2) - (\sum Sim_i)^2}} \right)^2 \dots \dots \dots (5-2)$$

where  $Obs_i$ ,  $Sim_i$  and  $N$  are defined as above in Eq. 5.1. Table 5.1 provides quantitative tidal performance of the various simulations. Coefficient of determination ( $R$  square) measures the variance that is explained by the model, which is the reduction of variance when using the model.  $R$  square ranges from 0 to 1; the model has strong predictive power when it is close to 1 and does not explain anything when it is close to 0.



Table 5.1

*RMSEs and  $R^2$  Values for Day 1 and 2 Simulations*

Day	Analyzed Data	Measure of Performance	SIM1	SIM2	SIM3	SIM4	SIM5
17-Jun-09	Velocity Magnitude	RMSE (m/sec)	0.2259	0.2335	0.2101	0.2055	0.1907
		$R^2$ (%)	12.97	10.06	14.16	22.70	21.22
18-Jun-09	Velocity Magnitude	RMSE (m/sec)	0.2356	0.2458	0.1908	0.2112	0.1886
		$R^2$ (%)	21.32	11.07	23.03	12.38	22.52

First, we note that the RMS errors shown in the last two columns of Table 5.1 (SIM 4 and SIM 5) are less than their corresponding simulations SIM 1 and SIM 3, demonstrating that the modified hydrodynamic model improved the simulations for both days (deployments). This is because an increase in Manning's  $n$  values in the ebb direction has a dampening effect on the velocity magnitude which the model over-predicted during initial simulations (SIM1 and SIM3). The best-performing simulation was SIM5 with RMSE values of 0.19 m/s and  $R^2$  values of 22%. It is notable the reduction of RMSE between SIMs 4 and 5 (as low as 0.19 m/s) and SIMs 1, 2 and 3 (as high as 0.25 m/s). Clearly, the data assimilation of the directionally variant bottom roughness formulation improves the model's ability to reproduce velocities in the LSJR, as measured by the data. Next, the calibrated model (i.e., SIM5) was run again, with the model results compared with a different dataset, to see that the directionally variant bottom roughness formulation performs optimally for a dataset other than the measured velocities.

## Hydrodynamic Model Validation

Model validation consisted of validating water surface elevations and streamflow. The importance of the model validation is to run the modified numerical code for comparison of model result to an observation dataset other than the tidal currents that were used to develop the modified numerical code. The validation period included May 01 – August 01, 2009 for streamflow data and June 16 – July 31, 2009 for water surface elevation data. Two tide gaging stations provided data for streamflow and water surface elevation near the Acosta Bridge (40 river km) and Dames Point (20 river km), respectively. Measurements of daily flows were obtained for the USGS Station 02246500 Jacksonville station (USGS, 2014) near the Acosta Bridge (<http://waterdata.usgs.gov/nwis>, 2015). A fixed tide gauge was used to measure water elevation near Dames Point (see Chapter 2).

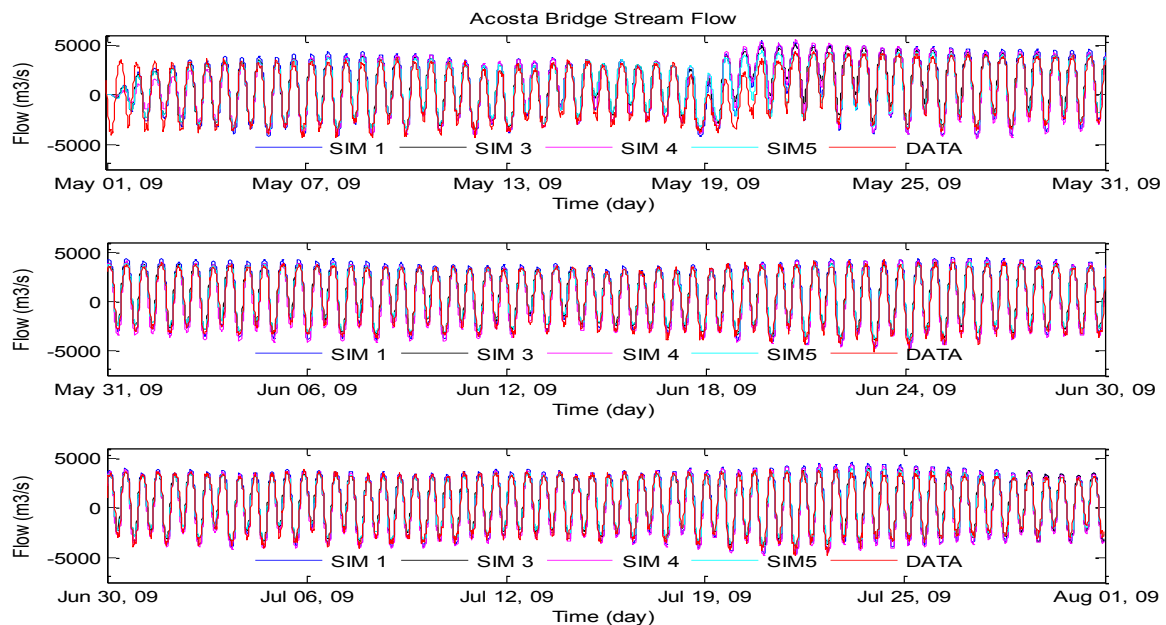


Figure 5.1. Validation plot, Acosta Bridge stream flow.

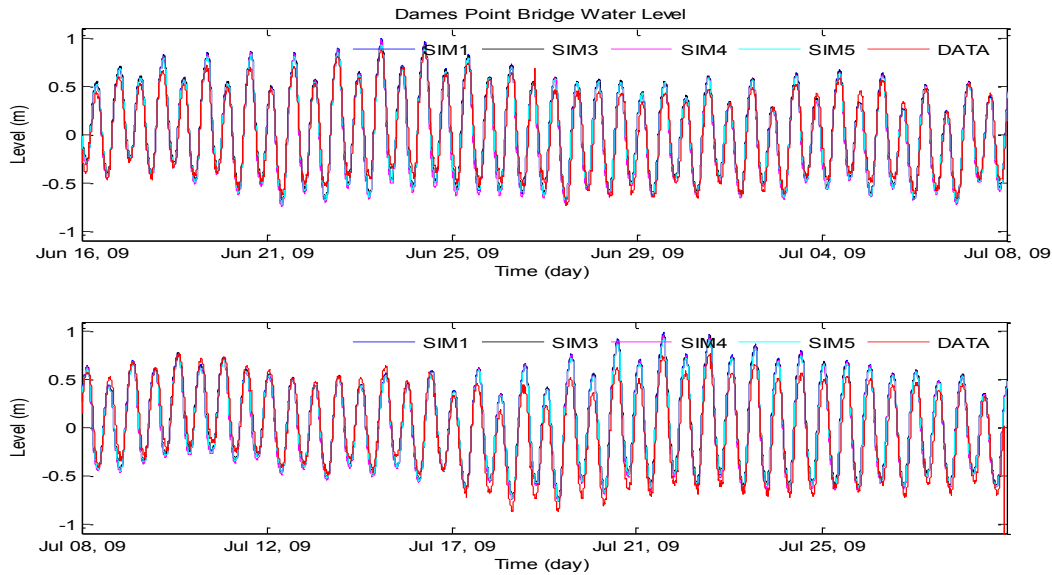


Figure 5.2. Validation plot, Dames Point Bridge water elevation.

Figure 5.1 and 5.2 shows time-series validation plots of the streamflow and water surface elevation at the Acosta Bridge and Dames Point, respectively. The validation results demonstrate a faithful reproduction of water levels and streamflow by the modified ADCIRC hydrodynamic model. There is a noticeable improvement in the model-data fit for SIM 5, namely in better predicting semidiurnal fluctuation in streamflow and in better predicting high water and low water of the tide.

Table 5.2

*RMSEs and  $R^2$  Values for Model Validation*

Duration	Analyzed Data	Measure of Performance	SIM1	SIM3	SIM4	SIM5
05/01/2009-08/01/2009	Stream flow (Acosta Bridge)	RMSE ( $\text{m}^3/\text{sec}$ )	790.17	841.68	886.86	728.96
		$R^2$ (%)	93.84	91.89	90.17	93.35
06/16/2009-07/29/2009	Tide level (Dames Point Bridge)	RMSE ( $\text{m}^3/\text{sec}$ )	0.2356	0.2378	0.2458	0.1908
		$R^2$ (%)	74.56	75.22	74.22	75.24

Root-mean-square error and coefficient of determination between observations and model results were computed for both stations and presented in Table 5.2. Again, SIM5 performed optimally with RMSE of 729 m<sup>3</sup>/s for streamflow and with RMSE of 0.19 m for water surface elevations, with notable increase in model performance when compared with the original (isotropic) bottom friction formulations (i.e., SIMs 1 and 3).

## CHAPTER 6: CONCLUSIONS

The two-dimensional, depth-integrated (2DDI) ADvanced CIRCulation (ADCIRC) model was updated for numerical simulation and data assimilation for hydrodynamics in the lower St. Johns River using a directionally variant bottom friction formulation. A subroutine in 2DDI ADCIRC, `nodalattr.F`, was updated to account for the anisotropy of bottom stress by enabling bottom roughness to vary in twelve directions. The directionally variant bottom friction formulation is guided by assimilating a velocity dataset that informs how to prescribe a vector-based Manning's  $n$  field for the lower St. Johns River. The modified hydrodynamic model was set up to compute hydrodynamics for flood and ebb phases of the tidal cycle, where the model incorporating spatially and temporally variable bottom roughness was tested against the original ADCIRC simulation output and observational data. Naturally, the calibrated model (i.e., using the directionally variant bottom roughness formulation) performed the best when compared with the velocity dataset used in the data assimilation. Though, the directionally variant bottom roughness formulation was shown to perform optimally for two other datasets (different than the velocity dataset used for data assimilation).

The model not only predicted the tidal measurements satisfactorily, but it also provided good estimates of actual bottom roughness. The bottom roughness factors were higher for the ebb direction when compared to the friction factors obtained for flood direction, which was in accordance with the tidal current data. In this sense, the calibration technique demonstrated here has applicability for parameter estimation associated with bottom roughness. With the absence of significant progress in understanding the underlying science of bottom roughness,

comprehensive and/or data assimilation-based calibration of hydrodynamic modeling is a practical solution, as demonstrated here.

As an alternative implementation, data assimilation can be performed iteratively via successive simulations and data assimilation to produce a converging result of the Manning's  $n$  value that represents 'the best' estimate of the actual bottom roughness. Such parameter estimation has utility in reducing the uncertainty dealing with parameterization of bottom roughness in future hydrodynamic modeling.

## REFERENCES

- ADCIRC, 2014. *ADvanced CIRCulation*. [Accessed online January 15, 2014: <http://adcirc.org/>.]
- Anderson, W. and Goolsby, D.A., 1973. *Flow and Chemical Characteristics of the St. Johns River at Jacksonville, Florida*. Information Circular 82. Tallahassee, FL: U.S. Geological Survey.
- Arcement, G.J., and Schneider, V.R., 1989. *Guide for selecting Manning's roughness coefficients for natural channels and flood plains*. U.S. Geological Survey Water-Supply Paper 2339, p. 38.
- Arcement, G.J., and Schneider, V.R. 2009. *Guide for Selecting Manning's Roughness Coefficients for Natural Channels and Flood Plains*. United States Geological Survey Water-supply Paper 2339.
- Arora & Bhaskaran. *Parameterization of bottom friction under combined wave-tide action in the Hooghly estuary, India*. Department of Ocean Engineering and Naval Architecture, Indian Institute of Technology Kharagpur, Kharagpur-721 302, India: Open University Press, 2003.
- Atkinson, J., Roberts, H., C. Hagen, S., Zou, S., Bacopoulos, P., Medeiros, S. . . . Cobell, Z. (n.d.). *Deriving frictional parameters and performing historical validation for an ADCIRC storm surge model of the Florida gulf coast*. Retrieved March 12, 2015, from <http://www.champs.cecs.ucf.edu/Publications/Refereed/Atkinsonetal.pdf>
- Bacopoulos, P., Funakoshi, Y., Hagen, S.C., Cox, A.T., and Cardone, V.J., 2009. *The role of meteorological forcing on the St. Johns River (Northeastern Florida)*. *Journal of Hydrology*, 369(1-2), 55-70.

- Bacopoulos, P., Hagen, S.C., Cox, A.T., Dally, W.R., Bratos, S., 2012. *Observation and simulation of winds and hydrodynamics in St. Johns and Nassau Rivers*. Journal of Hydrology 420–421, 391–402.
- Bacopoulos, P., Tang, Y., Wang, D., Hagen, S. C., and Demissie, H. K., 2015. *Integrated hydrologic-hydrodynamic modeling of flooding in the lower St. Johns River Basin caused by Tropical Storm Fay (2008)*. Journal of Hydrology, in review.
- Barnes, H.H., Jr., 1967. *Roughness characteristics of natural channels*. U.S. Geological Survey Water-Supply Paper 1849, 213 p.
- Buschman, F., 2011. *Flow and sediment transport in an Indonesian tidal network*. Utrecht: Department of Physical Geography, Faculty of Geosciences, Utrecht University.
- Bertin, X., Fortunato, A., and Oliveira, A., 2009. *A modeling-based analysis of processes driving wave-dominated inlets*. Cont.ShelfRes.29, 819–834.
- Bruun, P., 1978. *Stability of Tidal Inlets*. Theory and Engineering. Elsevier.
- Chai, T., and Draxler, R. R., 2014. *Root mean square error (RMSE) or mean absolute error (MAE)?* Geoscientific Model Development Discussions, 7, 1525-1534.
- Chow, V.T., 1959. *Open-channel hydraulics*. New York, McGraw- Hill Book Co., 680 p.
- Connor, J., and Brebbia, C.A., 1976. *Finite Element Techniques for Fluid Flow*. Newnes- Butterworth.
- Cowan, W.L., 1956. *Estimating hydraulic roughness coefficients*. Agricultural Engineering, v. 37, no. 7, p. 473-475.
- Dean, R.G. and Dalrymple, R.A., 1991. *Water Wave Mechanics for Engineering and Scientists*. World Scientific Press, 261-282.



- Dodet, G., Bertin, X., Bruneau, N., Fortunato, A., Nahon, A., and Roland, A., 2013. *Wave- current interactions in a wave-dominated tidal inlet*. J. Geophys. Res. 118, 1587–1605.
- Doos, K., Nycander, J., and Sigry, P., 2004. *Slope-dependent friction in a barotropic model*. J. Geophys. Res. 109, C01008.
- Garrett, J.R., 1977. *Review of drag coefficients over oceans and continents*. Monthly Weather Review 105, 915–929.
- Henderson, F.M., 1966. *Open-channel flow*. New York, MacMillan Publishing Co., Inc., 522 p.
- Kolar, R.L., Westerink, J.J., Cantekin, M.E., Blain, C.A., 1994. Aspects of nonlinear simulations using shallow-water models based on the wave continuity equation. Computers and Fluids 23, 523–538.
- Kagan, B., Sofina, E., and Rashidi, E. (n.d.). *The impact of the spatial variability in bottom roughness on tidal dynamics and energetics, a case study: The M 2 surface tide in the North European Basin*. Ocean Dynamics, 1425-1442.
- Kinnmark, I.P.E., 1985. *The Shallow Water Wave Equations*. Formulation, Analysis and Application. Lecture Notes in Engineering, vol. 15. Springer-Verlag, New York 187 pp.
- Kolar, R.L., Westerink, J.J., Cantekin, M.E., and Blain, C.A., 1994. *Aspects of nonlinear simulations using shallow-water models based on the wave continuity equation*. Comput. Fluids, 523–538.
- Kubatko, E.J., Dawson, C., and Westerink, J.J., 2008. *Time step restrictions for Runge-Kutta discontinuous Galerkin methods on triangular grids*. Journal of Computational Physics 227, 9697–9710.

- Kubatko, E.J., Westerink, J.J., and Dawson, C., 2006. *HP discontinuous Galerkin methods for advection-dominated problems in shallow water flow*. Computer Methods in Applied Mechanics and Engineering 196, 437–451.
- Kubatko, E.J., Bunya, S., Dawson, C., and Westerink, J.J., 2009. *Dynamic 762 p-adaptive Runge-Kutta discontinuous Galerkin methods for the shallow water equations*. Computer Methods in Applied Mechanics and Engineering 198, 1766–1774.
- Legates, D. R., and McCabe, G. J., 1999. *Evaluating the use of “goodness-of-fit” measures in hydrologic and hydroclimatic model validation*. Water resources research, 35(1), 233-241.
- Luettich Jr., R.A., Westerink, J.J., and Scheffner, N.W., 1992. *ADCIRC – An Advanced Three-dimensional Circulation Model for Shelves, Coasts, and Estuaries, I –Theory and Methodology of ADCIRC-2DDI and ADCIRC-3DL*. Technical Report DRP-92-6. US Army Corps of Engineers, Waterways Experiment Station, Vicksburg, Mississippi.
- Luettich, R. A., & Westerink, J. J. (2004). *Formulation and numerical implementation of the 2D/3D ADCIRC finite element model version 44. XX*. R. Luettich.
- Luettich, R.A., Jr., Westerink, J.J., 2006a. *ADCIRC: A Parallel Advanced Circulation Model for Oceanic, Coastal and Estuarine Waters; User’s Manual for Version 45.08*.  
<[http://adcirc.org/document/ADCIRC\\_title\\_page.html](http://adcirc.org/document/ADCIRC_title_page.html)>.
- Luettich, R.A., Jr., and Westerink, J.J., 2006b. *Formulation and Numerical Implementation of the 2D/3D ADCIRC Finite Element Model, Version 44.XX*. <[http://adcirc.org/document/ADCIRC\\_title\\_page.html](http://adcirc.org/document/ADCIRC_title_page.html)>.

- Lynch, D.R., and Gray, W.G., 1979. *A wave equation model for finite element tidal computations*. Comput. Fluids, 207–228.
- Malhadas, M., Leitaó, P., Silva, A., and Neves, R., 2009. *Effect of coastal waves on sea level in Obidos Lagoon, Portugal*. Cont. Shelf Res. 29, 1240–1250.
- Mallet, C., Howa, H., Garlan, T., Sottolichio, A., and Le, H. P., 2000. *Residual transport model in correlation with sedimentary dynamics over an elongated tidal sandbar in the Gironde estuary (Southwest France)*. Journal of Sedimentary Research, 1005–1016.
- Mayo, T., Butler, T., Dawson, C., and Hoteit, I., 2014. *Data assimilation within the Advanced Circulation (ADCIRC) modeling framework for the estimation of Manning's friction coefficient*. Ocean Modeling, 43-58
- Medeiros, S.C., Hagen, S.C., Chaouch, H., Feyen, J.C., Temimi, M., Weishampel, J.F., Funakoshi, Y., and Khanbilvardi, R., 2013. *Assessing the performance of a Northern Gulf of Mexico tidal model using satellite imagery*. Remote Sensing 5, 5662–5679.
- Nathan, L. D., 2007. *Hydrodynamic modeling of a hypothetical river*. Louisiana State University, Department of Civil and Environmental Engineering. Retrieved from [http://etd.lsu.edu/docs/available/etd-06142007-084318/unrestricted/Dill\\_thesis.pdf](http://etd.lsu.edu/docs/available/etd-06142007-084318/unrestricted/Dill_thesis.pdf)
- O'Brien, M.P., 1931. *Estuary tidal prism related to entrance areas*. CivilEng. 1 (8), 738–739.
- Olabarrieta, M., Warner, J., and Kumar, N., 2011. *Wave–current interaction in Willapa Bay*. J. Geophys. Res. 116, C12014, <http://dx.doi.org/10.1029/2011JC007387>.
- Pugh, D.T., 1987. *Tides, surges and mean sea-level*. A handbook for engineers and scientists. New York: John Wiley and Sons, 486 p.

- Reid, R.O. and K. Kaijura. 1957. *The Damping of Gravity Waves over a Permeable Seabed*.  
Trans. Amer. Geophysics. Union, 38.
- Ridderinkhof, H., 1988. *Tidal and residual flows in the western Dutch Wadden Sea, I: Numerical model and results*. Netherlands Journal of Sea Research, 1–21
- Simons, D.B., and Richardson, E.V., 1966. *Resistance to flow in alluvial channels*. U.S. Geological Survey Professional Paper 422-J, 61 p.
- Stevenson, T., 1886. *Design and Construction of Harbours*, third Ed. Adam and Charles Black, Edinburgh.
- Sucsy, P.V. and F. Morris. 2002. *Calibration of a Three-dimensional Circulation and Mixing Model of the Lower St. Johns River*. St. Johns River Water Management District Water Resources Department Technical Memorandum.
- Surfbreak Engineering Sciences, Inc., 2009. *Field Measurement Project in Support of the United States Army Corps of Engineers' Jacksonville Harbor Navigation Project, Duval County, Florida*. Report Submitted to Taylor Engineering, Inc. and US Army Corps of Engineers, Jacksonville District.
- Toth, D.J. 1993. *The Lower St. Johns River Basin Reconnaissance*. Hydrogeology. Technical Report SJ93-7, Volume 1, St. Johns River Water Management District, Palatka, Florida.
- USGS., 2014. *Water data for the nation*. [Accessed online March 7, 2014:  
<http://waterdata.usgs.gov/nwis>.]
- Van, S.M., and Swart, H.E., 2002. *Intermediate modeling of tidal inlet : spatial asymmetries in flow and mean sediment transport*. Continental Shelf Research, 1795–1810.

- Westerink, J.J., Luettich, R.A., Feyen, J.C., Atkinson, J.H., Dawson, C.N., Roberts, H.J., Powell, M.D., Dunion, J.P., Kubatko, E.J., and Pourtaheri, H., 2008. *A basin to channel scale unstructured grid hurricane storm surge model applied to southern Louisiana*. Monthly Weather Review 136, 833–864.
- Westerink, J.J., Luettich Jr., R.A., and Muccino, J.C., 1994. *Modeling tides in the Western North Atlantic using unstructured graded grids*. Tellus 46A.
- Zhang, J., Lu, X., Wang, P., and Wang, Y.P., 2010. *Study on linear and nonlinear bottom friction parameterizations for regional tidal models using data assimilation*. Continental Shelf Research. Retrieved, from [www.elsevier.com/locate/csr](http://www.elsevier.com/locate/csr).

## APPENDIX

## Appendix A: Modified Nodalattr.F file

All substantive code changes are highlighted in yellow.

```

C*****
**
C   PADCIRC VERSION 46.00 xx/xx/2006
C   last changes in this file VERSION 46.00
C
C   Written for ADCIRC v46.00 by Jason G. Fleming.
C*****
**
C
C-----
C-----
C   M O D U L E   N O D A L   A T T R I B U T E S
C-----
C
C   jgf46.00 This module manages nodal attribute data, including
C   bottom friction, tau0, startdry, directional wind speed reduction,
C   and etc. Will read the Nodal Attributes File (unit 13) and
C   initialize the nodal attribute arrays.
C
C   Handling data by label rather than an integer encoding should
C   result in increased transparency as well as ease the transition to
C   HDF5/NetCDF i/o. The labels were chosen according to the
C   guidelines of the CF Standard. Creating labels according to CF
C   Standard Guidelines should enhance interoperability with other
C   simulation frameworks.
C
C   To use a nodal attribute contained in the fort.13 file, the
C   corresponding attribute name must appear in the fort.15 file. A
C   list of nodal attributes is read in from the fort.15 file if the
C   fort.15 parameter NWP > 0. This also signals ADCIRC to look for a
C   fort.13 file.
C
C   Summary of the file format for the Nodal Attributes File:
C
C   AGRID                                ! user's comment line - should be a cross
C                                         !   reference to the grid file
C   NumOfNodes                          ! number of nodes, must match NP
C                                         !   from grid file
C   NAttr                               ! number of attributes contained in this
file
C

```

```

C      do i=1, NAttr
C          AttrName(i)          ! nodal attribute name (see
C                               !   valid names below)
C          Units(i)             ! physical units (ft, m/s, none)
C          ValuesPerNode(i)     ! number of values at each node for
C                               !   a particular attribute
C          DefaultVal(i)       ! default value(s) for the nodal attribute
C      end do
C
C      do i=1, NAttr
C          AttrName(i)          ! label of the attribute, again
C          NumNodesNotDefaultVal(i) ! number of nodes with non-default values
C          do j=1, NumNodesNotDefault(i)
C              n, (Attr(n,k), k=1,ValuesPerNode(i))
C          end do
C      end do
C
C
C      Valid labels are as follows:
C
C      ADCIRC Variable:      CF-Style Label:
C      Tau0                  "primitive_weighting_in_continuity_equation"
C      StartDry              "surface_submergence_state"
C      Fric                  "quadratic_friction_coefficient_at_sea_floor"
C      z0Land                "surface_directional_effective_roughness_length"
C                           (z0Land has ValuesPerNode = 12)
C      VCanopy               "surface_canopy_coefficient"
C      BK,BAlpha,BDelX,POAN "bridge_pilings_friction_parameters"
C                           (bridge_pilings... has ValuesPerNode=4)
C      ManningsN             "mannings_n_at_sea_floor"
C      Chezy                 "chezy_friction_coefficient_at_sea_floor"
C      Z0b_var               "bottom_roughness_length"
C      GeoidOffset           "sea_surface_height_above_geoid"
C      EVM                   "average_horizontal_eddy_viscosity_in_sea_water_wrt_depth"
C      EVC                   "average_horizontal_eddy_diffusivity_in_sea_water_wrt_depth"
C      Tao0MinMax            "min_and_max_primitive_weighting_in_continuity_equation"
C
C-----
C      MODULE NodalAttributes
C      USE SIZES

```

Casey 100210: Allow SWAN to handle wave refraction as a nodal attribute.

```

C      I've placed these changes outside the #ifdef CSWAN flags
C      because we want to be able to use the same fort.13 files
C      for both ADCIRC and SWAN+ADCIRC runs. This way, the new
C      nodal attribute will be processed but only applied when

```

```

C          ADCIRC is coupled to SWAN.
LOGICAL          :: LoadSwanWaveRefrac
LOGICAL          :: FoundSwanWaveRefrac
CHARACTER(LEN=80) :: SwanWaveRefracUnits
INTEGER          :: SwanWaveRefracNoOfVals
REAL(SZ)         :: SwanWaveRefracDefVal
REAL(SZ),ALLOCATABLE :: SwanWaveRefrac(:)

C
Corbitt 120321: Allow Advection to be Turned on Locally instead of Globally
LOGICAL          :: LoadAdvectionState
LOGICAL          :: FoundAdvectionState
CHARACTER(LEN=80) :: AdvectionStateUnits
INTEGER          :: AdvectionStateNoOfVals
REAL(SZ)         :: AdvectionStateDefVal
REAL(SZ),ALLOCATABLE :: AdvectionState(:)

C
C   The following flags are .true. if the corresponding data are
C   required for the run, according to the unit 15 control file
LOGICAL LoadTau0
LOGICAL LoadStartDry
LOGICAL LoadDirEffRLen
LOGICAL LoadCanopyCoef
LOGICAL LoadQuadraticFric
LOGICAL LoadBridgePilings
LOGICAL LoadChezy
LOGICAL LoadManningsN
LOGICAL LoadGeoidOffset
LOGICAL LoadEVM
LOGICAL LoadEVC
LOGICAL LoadTau0MinMax
LOGICAL LoadZ0b_var
LOGICAL LoadEleSlopeLim ! zc: elemental slope limiter

C
C   The following flags are .true. if there are data with the
C   corresponding label in the unit 13 file.
LOGICAL FoundTau0
LOGICAL FoundStartDry
LOGICAL FoundDirEffRLen
LOGICAL FoundCanopyCoef
LOGICAL FoundQuadraticFric
LOGICAL FoundBridgePilings
LOGICAL FoundChezy
LOGICAL FoundManningsN
LOGICAL FoundGeoidOffset
LOGICAL FoundEVM
LOGICAL FoundEVC
LOGICAL FoundTau0MinMax
LOGICAL FoundZ0b_var

```



```

LOGICAL FoundEleSlopeLim
C
C These variables hold the strings which describe the attribute's
C units. These data are loaded from the file, but not used as of
C v46.00.
CHARACTER(len=80) Tau0Units
CHARACTER(len=80) StartDryUnits
CHARACTER(len=80) DirEffRLenUnits
CHARACTER(len=80) CanopyCoefUnits
CHARACTER(len=80) QuadraticFricUnits
CHARACTER(len=80) BridgePilingsUnits
CHARACTER(len=80) ChezyUnits
CHARACTER(len=80) ManningsNUnits
CHARACTER(len=80) GeoidOffsetUnits
CHARACTER(len=80) EVMUnits
CHARACTER(len=80) EVCUnits
CHARACTER(len=80) Tau0MinMaxUnits
CHARACTER(len=80) Z0b_varUnits
CHARACTER(len=80) EleSlopeLimUnits
C
C These variables hold the number of values per node for each
C attribute.
INTEGER Tau0NoOfVals
INTEGER StartDryNoOfVals
INTEGER DirEffRLenNoOfVals
INTEGER CanopyCoefNoOfVals
INTEGER QuadraticFricNoOfVals
INTEGER BridgePilingsNoOfVals
INTEGER ChezyNoOfVals
INTEGER ManningsNNoOfVals
INTEGER GeoidOffsetNoOfVals
INTEGER EVMNoOfVals
INTEGER EVCTNoOfVals
INTEGER Tau0MinMaxNoOfVals
INTEGER Z0b_varNoOfVals
INTEGER EleSlopeLimNoOfVals
C
C These variables hold the default values for each attribute.
REAL(SZ) Tau0DefVal
REAL(SZ) StartDryDefVal
REAL(SZ) DirEffRLenDefVal(12)
REAL(SZ) CanopyCoefDefVal
REAL(SZ) QuadraticFricDefVal
REAL(SZ) BridgePilingsDefVal(4)
REAL(SZ) ChezyDefVal
REAL(SZ) ManningsNDefVal
REAL(SZ) GeoidOffsetDefVal
REAL(SZ) EVMDfVal

```

```

REAL(SZ) EVCDefVal
REAL(SZ) Tau0MinMaxDefVal(2)
REAL(SZ) Z0b_varDefVal
REAL(SZ) EleSlopeLimDefVal
C
  INTEGER NumOfNodes      ! number of nodes listed in unit 13 file, cf. NP
  INTEGER NAttr           ! number of nodal attributes in the unit 13 file
C
C   The following variables are inputs from the unit 15 model param. file
  INTEGER NWP             ! number of nodal attributes to read from file
  INTEGER NoLiBF          ! nonlinear bottom friction indicator
  REAL(SZ) Tau0           ! primitive continuity eqn. weight
  REAL(SZ) Tau            ! linear friction coefficient (1/sec)
  REAL(SZ) CF             ! 2DDI bottom fric. coef., effect varies based on
NoLiBF
  REAL(SZ) HBreak ! break depth for NOLIBF .eq. 2
  REAL(SZ) FTheta ! dimless param. for NOLIBF .eq. 2
  REAL(SZ) FGamma ! dimless param. for NOLIBF .eq.
  REAL(SZ) ESLM   ! horizontal eddy viscosity (length^2/time)
  REAL(SZ) ESLC   ! horizontal eddy diffusivity (length^2/time)
  INTEGER IFLINBF ! flag to turn on linear bottom friction
  INTEGER IFNLBF  ! flag to turn on nonlinear bottom friction
  INTEGER IFHYBF  ! flag to turn on hybrid bottom friction
  REAL(SZ) BFCdLLimit ! lower limit of quadratic bottom friction
  REAL(SZ) Tau0FullDomainMin ! lower limit of tau0 if time varying
  REAL(SZ) Tau0FullDomainMax ! upper limit of tau0 if time varying
C   jgf48.42 Used to control the back-loaded time averaged tau0
  REAL(SZ), PARAMETER :: AlphaTau0 = 0.25d0
C
C   Nodal attributes.
  REAL(SZ), ALLOCATABLE :: STARTDRY(:) ! 1=nodes below geoid initially
dry
  REAL(SZ), ALLOCATABLE :: FRIC(:)      ! bottom friction coefficient
  REAL(SZ), ALLOCATABLE, TARGET :: TAU0VAR(:)! primitive equation
weighting
  REAL(SZ), ALLOCATABLE :: Tau0Temp(:) ! used in time varying tau0
C   jjw&sb46.39.sb01: Base (original) primitive equation weighting.
C   Tau0Var may be optimized later based on Tau0Base
  REAL(SZ), ALLOCATABLE :: TAU0BASE(:)
C   jgf47.33 Used for time averaged tau0
  REAL(SZ), ALLOCATABLE :: LastTau0(:)

C   jgf47.06: Added variables to trigger calculations and output of tau0
C   jgf47.30: Added "FullDomain" or "High Res Areas Only" distinction
C   jgf47.31: Added time averaging to time varying tau0
C   jgf48.42: Added backloaded time averaged tau0
  LOGICAL :: HighResTimeVaryingTau0 ! .true. if Tau0 == -3.x in
fort.15

```

```

        LOGICAL :: FullDomainTimeVaryingTau0 ! .true. if Tau0 == -5.x in
fort.15
        LOGICAL :: OutputTau0      ! .true. if Tau0Dig2 == -1 in fort.15
        LOGICAL :: TimeAveragedTau0 ! .true. if Tau0 == -6.x in fort.15
        LOGICAL :: BackLoadedTimeAveragedTau0 ! .true. if Tau0 == -7.x
        LOGICAL,ALLOCATABLE :: elemental_slope_limiter_active(:) ! .true. if
elemental_slope_limiter_grad_max
                                                    ! has been
exceeded, initialized as .false.
        LOGICAL,ALLOCATABLE :: elemental_slope_limiter_max_exceeded(:) ! .true.
if maximum gradient has been
                                                    !
exceeded, used for gradient warnings

        REAL(SZ), ALLOCATABLE :: z0land(:, :) ! directional wind speed red. fac.
        REAL(SZ), ALLOCATABLE :: vcanopy(:) ! canopy coefficient
C      The following attribute contains BK(I),BALPHA(I),BDELX(I), and POAN(I)
        REAL(SZ), ALLOCATABLE :: BridgePilings(:, :)
        REAL(SZ), ALLOCATABLE :: Chezy(:)
        REAL(SZ), ALLOCATABLE :: ManningsN(:)
        REAL(SZ), ALLOCATABLE :: GeoidOffset(:)
        REAL(SZ), ALLOCATABLE :: EVM(:)
        REAL(SZ), ALLOCATABLE :: EVC(:)
        REAL(SZ), ALLOCATABLE :: Tau0MinMax(:, :) ! (node,i); i=1(min),
i=2(max)
        REAL(SZ), ALLOCATABLE :: Z0b_var(:) ! patially varying 3D bottom
roughness length
        REAL(SZ), ALLOCATABLE :: elemental_slope_limiter_grad_max(:)
C
        INTEGER i      ! node loop counter
        INTEGER j      ! attribute values loop counter
        INTEGER k      ! attribute loop counter
C
C - - - - -
CONTAINS !- - - - -
C - - - - -

C      -----
C      S U B R O U T I N E      I N I T N A M O D U L E
C      -----
C
C      jgf46.00 Subroutine to initialize the variables in the nodal
C      attributes module.
C
C      -----
C      SUBROUTINE InitNAModule()
C      IMPLICIT NONE
C

```

Casey 100210: Make changes compact.

```
LoadSwanWaveRefrac      = .FALSE.
FoundSwanWaveRefrac     = .FALSE.
SwanWaveRefracNoOfVals  = 1
SwanWaveRefracDefVal    = 0.D0
```

Corbitt 120321:

```
LoadAdvectionState      = .False.
FoundAdvectionState     = .FALSE.
AdvectionStateNoOfVals  = 1
AdvectionStateDefVal    = 0.D0
```

C

```
LoadTau0                = .False.
LoadStartDry            = .False.
LoadDirEffRLen          = .False.
LoadManningsN           = .False.
LoadQuadraticFric       = .False.
LoadChezy               = .False.
LoadBridgePilings       = .False.
LoadCanopyCoef          = .False.
LoadGeoidOffset         = .False.
LoadEVM                 = .False.
LoadEVC                 = .False.
LoadTau0MinMax          = .False.
LoadZ0b_var             = .False.
LoadEleSlopeLim         = .False.
```

C

```
FoundTau0               = .False.
FoundStartDry           = .False.
FoundDirEffRLen         = .False.
FoundManningsN          = .False.
FoundQuadraticFric      = .False.
FoundChezy              = .False.
FoundBridgePilings      = .False.
FoundCanopyCoef         = .False.
FoundGeoidOffset        = .False.
FoundEVM                = .False.
FoundEVC                = .False.
FoundTau0MinMax         = .False.
FoundZ0b_var            = .False.
FoundEleSlopeLim        = .False.
```

C

C

```
Tau0NoOfVals            = 1
StartDryNoOfVals        = 1
DirEffRLenNoOfVals      = 12
QuadraticFricNoOfVals   = 1
ChezyNoOfVals           = 1
ManningsNNoOfVals       = 1
```

```

    BridgePilingsNoOfVals = 4
    CanopyCoefNoOfVals    = 1
    GeoidOffsetNoOfVals   = 1
    EVMNoOfVals           = 1
    EVCNoOfVals           = 1
    Tau0MinMaxNoOfVals    = 2
    Z0b_varNoOfVals       = 1
    EleSlopeLimNoOfVals   = 1
C
    Tau0DefVal             = 0.0
    StartDryDefVal         = 0.0
    DO j=1, DirEffRLenNoOfVals
        DirEffRLenDefVal(j) = 0.0
    END DO
    CanopyCoefDefVal        = 1.0 ! jgf49.1001 default is now full wind
stress
    QuadraticFricDefVal    = 0.0
    DO j=1, BridgePilingsNoOfVals
        BridgePilingsDefVal(j) = 0.0
    END DO
    ChezyDefVal            = 0.0
    ManningsNDefVal        = 0.0
    GeoidOffsetDefVal      = 0.0
    EVMDefVal              = 0.0
    EVCDefVal              = 0.0
    DO j=1, Tau0MinMaxNoOfVals
        Tau0DefVal = 0.0
    END DO
    Z0b_varDefVal          = 0.001
    EleSlopeLimDefVal      = 0D0
C
    HighResTimeVaryingTau0 = .False.
    FullDomainTimeVaryingTau0 = .False.
    OutputTau0              = .False.
    TimeAveragedTau0        = .False.
    BackLoadedTimeAveragedTau0 = .False.
C
    HBREAK=1.d0
    FTHETA=1.d0
    FGAMMA=1.d0
C
C kmd48.33bc this resets the ESLM to 0 and if using constant eddy
C viscosity it eliminates what the user specified in
C the input file. The InitNAModule originally come before
C the read_input call but now it appears after the read_input
C call.
!     ESLM=0.0
     ESLC=0.0

```

```

C      RETURN
C      -----
C      END SUBROUTINE InitNAModule
C      -----

C      -----
C      S U B R O U T I N E      R E A D   N O D A L   A T T R
C      -----
C
C      jgf46.00 Subroutine to read the nodal attributes file (unit 13).
C      -----
C      SUBROUTINE ReadNodalAttr(NScreen, ScreenUnit, MyProc, NAbOut)
C      IMPLICIT NONE
C      INTEGER, intent(in) :: NScreen ! nonzero for debug info to screen
C      INTEGER, intent(in) :: ScreenUnit ! i/o for screen
C      INTEGER, intent(in) :: MyProc ! in parallel, only MyProc=0 i/o to
screen
C      INTEGER, intent(in) :: NAbOut ! 1 to abbrev. output to unit 16
C
C      LOGICAL NAFound ! .true. if Nodal Attributes File (fort.13) exists
C      INTEGER ErrorIO ! zero if file opened successfully
C      CHARACTER(len=80) AttrName ! string where the attribute name is stored
C      CHARACTER(len=80) header ! string where alphanumeric file id is
stored
C      INTEGER NumNodesNotDefault ! number of individual nodes to specify
C      LOGICAL SkipDataSet ! .true. if a data set in unit 13 is not needed
C      CHARACTER(len=80) Skipped ! data in unit 13 we do not need
C      INTEGER L ! line counter
C
C      ! temp array; used to load a real from the file,
C      ! then convert to integer
C      REAL(sz), ALLOCATABLE :: real_loader(:)
C
C      NAFound = .False.
C      SkipDataSet = .False.
C
C      Check to make sure that NWP is a valid number.
C      IF (NWP.LT.0) THEN
C        IF(NSCREEN.NE.0.AND.MYPROC.EQ.0) THEN
C          WRITE(ScreenUnit,9972)
C          WRITE(ScreenUnit,*) 'NWP =',NWP
C          WRITE(ScreenUnit,9728)
C          WRITE(ScreenUnit,9973)
C        ENDIF

```

```

        WRITE(16,9972)
        WRITE(16,*) 'NWP =',NWP
        WRITE(16,9728)
        WRITE(16,9973)
9728    FORMAT(/,1X,'Your selection of NWP (a UNIT 15 input ',
&        'parameter) is not an allowable value')
        STOP                ! We're toast.
    ENDIF
C
C    Check to see if there are nodal attributes to be read in. If not,
C    simply return.
    IF (NWP.EQ.0) THEN
        WRITE(16,231) NWP
231    FORMAT(/,5X,'NWP = ',I2,
&        /,9X,'A Nodal Attributes File (unit 13)',
&        /,9X,'will not be used.')
        RETURN
    ENDIF
C
C    Otherwise, get on with it.
    WRITE(16,232) NWP
232    FORMAT(/,5X,'NWP = ',I2,
&        /,9X,'Must read Nodal Attributes File (unit 13).')
C
C    Determine if the Nodal Attributes File exists.
    INQUIRE(FILE=TRIM(INPUTDIR)//'/'/'fort.13',EXIST=NAFound)
C
    IF (.not.NAFound) THEN
        WRITE(16,1001)      ! Nodal Attributes file
        WRITE(16,1011)      ! was not found.
        WRITE(16,9973)      ! execution terminated
        IF (NScreen.ne.0.and.MyProc.eq.0) THEN
            WRITE(ScreenUnit,1001)
            WRITE(ScreenUnit,1011)
            WRITE(ScreenUnit,9973)      ! execution terminated
        ENDIF
        STOP
    ENDIF
C
C    Read the unit 15 control file to determine what data must be
C    loaded from nodal attributes file.
    WRITE(16,235) NWP
235    FORMAT(/,9X,'Need to load ',I2,' nodal attribute(s):')
    DO k=1,NWP
        READ(15,*) AttrName
        WRITE(16,'(14X,A80)') AttrName
        SELECT CASE (AttrName)
            CASE("primitive_weighting_in_continuity_equation")

```

```

        LoadTau0 = .True.
CASE("surface_submergence_state")
    LoadStartDry = .True.
CASE("quadratic_friction_coefficient_at_sea_floor")
    LoadQuadraticFric = .True.
CASE("surface_directional_effective_roughness_length")
    LoadDirEffRLen = .True.
CASE("surface_canopy_coefficient")
    LoadCanopyCoef = .True.
CASE("bridge_pilings_friction_parameters")
    LoadBridgePilings = .True.
CASE("mannings_n_at_sea_floor")
    LoadManningsN = .True.
CASE("chezy_friction_coefficient_at_sea_floor")
    LoadChezy = .True.
CASE("bottom_roughness_length")
    LoadZ0b_var = .True.
CASE("sea_surface_height_above_geoid")
    LoadGeoidOffset = .True.
CASE
& ("average_horizontal_eddy_viscosity_in_sea_water_wrt_depth")
    LoadEVM = .True.
CASE
& ("average_horizontal_eddy_diffusivity_in_sea_water_wrt_depth")
    LoadEVC = .True.
CASE
& ("min_and_max_primitive_weighting_in_continuity_equation")
    LoadTau0MinMax = .True.
Casey 100210: Allow SWAN to handle wave refraction as a nodal attribute.
CASE("wave_refraction_in_swan")
    LoadSwanWaveRefrac = .TRUE.
Corbitt 120321: Allow advection to be turned on locally instead of globally
CASE("advection_state")
    LoadAdvectionState = .True.
CASE("elemental_slope_limiter")
    LoadEleSlopeLim = .TRUE.
CASE DEFAULT
    WRITE(16,1000)          ! unit 15 Model Parameter file
    WRITE(16,1021) AttrName ! contains invalid name
    IF (NScreen.ne.0.and.MyProc.eq.0) THEN
        WRITE(ScreenUnit,1000)
        WRITE(ScreenUnit,1021) AttrName
    ENDIF
END SELECT
ENDDO
C
C Now open the nodal attributes (unit 13) file.
WRITE(16,240)

```



```

240  FORMAT(/,9X,'Nodal Attributes File (unit 13) was found.',
&      ' Opening file.')
      OPEN(UNIT=13, FILE=TRIM(INPUTDIR)//'/'/'/'/'fort.13',
&      IOSTAT=ErrorIO)
      IF ( ErrorIO .GT. 0 ) THEN
          WRITE(16,1001)          ! Nodal attribute file
          WRITE(16,1005)          ! exists but can't be opened
          WRITE(16,9973)          ! execution terminated
          IF (NScreen.ne.0.and.MyProc.eq.0) THEN
              WRITE(ScreenUnit,1001)
              WRITE(ScreenUnit,1005)
              WRITE(ScreenUnit,9973)
          ENDIF
          STOP                      ! We're toast.
      ENDIF
C
C      Read each attribute name, units, number of values, and default value
      READ(13,'(A80)') header
      WRITE(16,250)
250  FORMAT(/,9X,'User comment line from unit 13:')
      WRITE(16,'(14X,A80,/)' ) header
      READ(13,*) NumOfNodes        ! number of nodes according to unit 13
      READ(13,*) NAttr             ! number of attributes in the unit 13 file
      DO k=1, NAttr
          READ(13,*) AttrName
          WRITE(16,'(9X,A80)') AttrName
          WRITE(16,260)
260  FORMAT(14X,'was found!',/)
          SELECT CASE (AttrName)
              CASE("primitive_weighting_in_continuity_equation")
                  FoundTau0 = .True.
                  READ(13,'(A80)') Tau0Units
                  READ(13,*) Tau0NoOfVals
                  READ(13,*) Tau0DefVal
              CASE("surface_submergence_state")
                  FoundStartDry = .True.
                  READ(13,'(A80)') StartDryUnits
                  READ(13,*) StartDryNoOfVals
                  READ(13,*) StartDryDefVal
              CASE("quadratic_friction_coefficient_at_sea_floor")
                  FoundQuadraticFric = .True.
                  READ(13,'(A80)') QuadraticFricUnits
                  READ(13,*) QuadraticFricNoOfVals
                  READ(13,*) QuadraticFricDefVal
              CASE("surface_directional_effective_roughness_length")
                  FoundDirEffRLen = .True.
                  READ(13,'(A80)') DirEffRLenUnits
                  READ(13,*) DirEffRLenNoOfVals

```

```

        READ(13,*)
&      (DirEffRLenDefVal(j),j=1,DirEffRLenNoOfVals)
CASE("surface_canopy_coefficient")
    FoundCanopyCoef = .True.
    READ(13,'(A80)') CanopyCoefUnits
    READ(13,*) CanopyCoefNoOfVals
    READ(13,*) CanopyCoefDefVal
CASE("bridge_pilings_friction_parameters")
    FoundBridgePilings = .True.
    READ(13,'(A80)') BridgePilingsUnits
    READ(13,*) BridgePilingsNoOfVals
    READ(13,*)
&      (BridgePilingsDefVal(j),j=1,BridgePilingsNoOfVals)
CASE("mannings_n_at_sea_floor")
    FoundManningsN = .True.
    READ(13,'(A80)') ManningsNUnits
    READ(13,*) ManningsNNoOfVals
    READ(13,*) ManningsNDefVal
CASE("bottom_roughness_length")
    FoundZ0b_var = .True.
    READ(13,'(A80)') Z0b_varUnits
    READ(13,*) Z0b_varNoOfVals
    READ(13,*) Z0b_varDefVal
CASE("chezy_friction_coefficient_at_sea_floor")
    FoundChezy = .True.
    READ(13,'(A80)') ChezyUnits
    READ(13,*) ChezyNoOfVals
    READ(13,*) ChezyDefVal
CASE("sea_surface_height_above_geoid")
    FoundGeoidOffset = .True.
    READ(13,'(A80)') GeoidOffsetUnits
    READ(13,*) GeoidOffsetNoOfVals
    READ(13,*) GeoidOffsetDefVal
CASE
&      ("average_horizontal_eddy_viscosity_in_sea_water_wrt_depth")
    FoundEVM = .True.
    READ(13,'(A80)') EVMUnits
    READ(13,*) EVMNoOfVals
    READ(13,*) EVMDefVal
CASE
&      ("average_horizontal_eddy_diffusivity_in_sea_water_wrt_depth")
    READ(13,'(A80)') EVCUnits
    READ(13,*) EVNoOfVals
    READ(13,*) EVDefVal
CASE
&      ("min_and_max_primitive_weighting_in_continuity_equation")
    FoundTau0MinMax = .True.
    READ(13,'(A80)') Tau0MinMaxUnits

```

```

        READ(13,*) Tau0MinMaxNoOfVals
        READ(13,*) (Tau0MinMaxDefVal(j),j=1,Tau0MinMaxNoOfVals)
Casey 100210: Allow SWAN to handle wave refraction as a nodal attribute.
        CASE("wave_refraction_in_swan")
            FoundSwanWaveRefrac = .TRUE.
            READ(13,'(A80)') SwanWaveRefracUnits
            READ(13,*) SwanWaveRefracNoOfVals
            READ(13,*) SwanWaveRefracDefVal
Corbitt 120321: Allow advection to be turned on locally instead of globally
        CASE("advection_state")
            FoundAdvectionState = .TRUE.
            READ(13,'(A80)') AdvectionStateUnits
            READ(13,*) AdvectionStateNoOfVals
            READ(13,*) AdvectionStateDefVal
        CASE("elemental_slope_limiter")
            FoundEleSlopeLim = .TRUE.
            READ(13,'(A80)') EleSlopeLimUnits
            READ(13,*) EleSlopeLimNoOfVals
            READ(13,*) EleSlopeLimDefVal
        CASE DEFAULT
            WRITE(16,1001)          ! Nodal Attributes file
            WRITE(16,1021) AttrName ! contains invalid name
            IF (NScreen.ne.0.and.MyProc.eq.0) THEN
                WRITE(ScreenUnit,1001)
                WRITE(ScreenUnit,1021) AttrName
            ENDIF
            READ(13,'(A80)') Skipped ! skip the Units for the invalid name
            READ(13,'(A80)') Skipped ! skip the NoOfVals for invalid name
        END SELECT
    END DO

C
C   Determine if there are any attributes required by the fort.15 file
C   that are not in the nodal attributes file.
    IF(((LoadTau0).and.(.not.FoundTau0)).or.
& ((LoadStartDry).and.(.not.FoundStartDry)).or.
& ((LoadQuadraticFric).and.
& (.not.FoundQuadraticFric)).or.
& ((LoadDirEffRLen).and.
& (.not.FoundDirEffRLen)).or.
& ((LoadCanopyCoef).and.
& (.not.FoundCanopyCoef)).or.
& ((LoadBridgePilings).and.
& (.not.FoundBridgePilings)).or.
& ((LoadManningsN).and.
& (.not.FoundManningsN)).or.
& ((LoadZ0b_var).and.
& (.not.FoundZ0b_var)).or.
& ((LoadGeoidOffset).and.

```

```

&    (.not.FoundGeoidOffset)).or.
&    ((LoadChezy).and.(.not.FoundChezy)).or.
&    ((LoadEVM).and.(.not.FoundEVM)).or.
&    ((LoadEVC).and.(.not.FoundEVC)).or.
Casey 100210: Allow SWAN to handle wave refraction as a nodal attribute.
&    ((LoadSwanWaveRefrac).and.(.not.FoundSwanWaveRefrac)).or.
Corbitt 120321: Allow advection to be turned on locally instead of globally
&    ((LoadAdvectionState).and.(.not.FoundAdvectionState)).or.
&    ((LoadEleSlopeLim).and.(.not.FoundEleSlopeLim)).or.
&    ((LoadTau0MinMax).and.(.not.FoundTau0MinMax)) THEN
    WRITE(16,1111)
1111    FORMAT('ERROR: Nodal Attributes file (unit 13) does '
&            'not contain all the attributes listed in the '
&            /,'model parameter file (unit 15).')
    WRITE(16,9973)          ! execution terminated
    IF (NScreen.ne.0.and.MyProc.eq.0) THEN
        WRITE(ScreenUnit,1111)
        WRITE(ScreenUnit,9973)          ! execution terminated
    ENDIF
    STOP                          ! We're toast.
ENDIF

C    Allocate memory to hold our data.
    ALLOCATE(TAU0VAR(NumOfNodes),TAU0BASE(NumOfNodes)) ! jjw&sb46.39sb01
    ALLOCATE(STARTDRY(NumOfNodes))
    ALLOCATE(FRIC(NumOfNodes))
    ALLOCATE(z0land(NumOfNodes,DirEffRLenNoOfVals))
    ALLOCATE(vcanopy(NumOfNodes))
    ALLOCATE(BridgePilings(NumOfNodes,BridgePilingsNoOfVals))
    ALLOCATE(GeoidOffset(NumOfNodes))
    ALLOCATE(Chezy(NumOfNodes))
    ALLOCATE(ManningsN(NumOfNodes))
    ALLOCATE(Z0b_var(NumOfNodes))
    ALLOCATE(EVM(NumOfNodes))
    ALLOCATE(EVC(NumOfNodes))
    ALLOCATE(Tau0MinMax(NumOfNodes,Tau0MinMaxNoOfVals))
Casey 100210: Allow SWAN to handle wave refraction as a nodal attribute.
    ALLOCATE(SwanWaveRefrac(NumOfNodes))
Corbitt 120321: Allow advection to be turned on locally instead of globally
    ALLOCATE(AdvectionState(NumOfNodes))
    ALLOCATE(elemental_slope_limiter_grad_max(NumOfNodes))
    ALLOCATE(elemental_slope_limiter_active(NumOfNodes))
    ALLOCATE(elemental_slope_limiter_max_exceeded(NumOfNodes))
    elemental_slope_limiter_active(:) = .FALSE.  !...Alloate and initialize
to keep
                                                !    track of
elemental_slope_limiter nodes
    elemental_slope_limiter_max_exceeded(:) = .FALSE.

```

```

C
C   Now read each of the attributes required by the model parameter
C   (unit 15) file and skip past the others.
      WRITE(16,270) NWP
270  FORMAT(/,9X,'Now reading ',I2,' nodal attribute(s).')
      DO k=1, NAttr
        WRITE(16,280) k
280  FORMAT(/,9X,'Attribute ',I2,':')
        READ(13,*) AttrName
        READ(13,*) NumNodesNotDefault
        WRITE(16,'(14X,A80)') AttrName
        SELECT CASE (AttrName)
          CASE("primitive_weighting_in_continuity_equation")
            IF (LoadTau0) THEN
              CALL LoadAttrVec(TAU0VAR, Tau0DefVal,
&                NumNodesNotDefault, NScreen, MyProc, NAbOut)
            ELSE
              SkipDataSet = .True.
            ENDIF
          CASE("surface_submergence_state")
            IF (LoadStartDry) THEN
              CALL LoadAttrVec(STARTDRY, StartDryDefVal,
&                NumNodesNotDefault, NScreen, MyProc, NAbOut)
            ELSE
              SkipDataSet = .True.
            ENDIF
          CASE("quadratic_friction_coefficient_at_sea_floor")
            IF (LoadQuadraticFric) THEN
              CALL LoadAttrVec(FRIC, QuadraticFricDefVal,
&                NumNodesNotDefault, NScreen, MyProc, NAbOut)
            ELSE
              SkipDataSet = .True.
            ENDIF
          CASE("surface_directional_effective_roughness_length")
            IF (LoadDirEffRLen) THEN
              CALL LoadAttrMat(z0land, DirEffRLenNoOfVals,
&                DirEffRLenDefVal, NumNodesNotDefault,
&                NScreen, MyProc, NAbOut)
            ELSE
              SkipDataSet = .True.
            ENDIF
          CASE("surface_canopy_coefficient")
            IF (LoadCanopyCoef) THEN
              CALL LoadAttrVec(vcanopy, CanopyCoefDefVal,
&                NumNodesNotDefault, NScreen, MyProc, NAbOut)
            ELSE
              SkipDataSet = .True.
            ENDIF

```

```

CASE("bridge_pilings_friction_parameters")
  IF (LoadBridgePilings) THEN
    CALL LoadAttrMat(BridgePilings, BridgePilingsNoOfVals,
&      BridgePilingsDefVal, NumNodesNotDefault,
&      NScreen, MyProc, NAbOut)
  ELSE
    SkipDataSet = .True.
  ENDIF
CASE("mannings_n_at_sea_floor")
  IF (LoadManningsN) THEN
    CALL LoadAttrVec(ManningsN, ManningsNDefVal,
&      NumNodesNotDefault, NScreen, MyProc, NAbOut)
  ELSE
    SkipDataSet = .True.
  ENDIF
CASE("bottom_roughness_length")
  IF (LoadZ0b_var) THEN
    CALL LoadAttrVec(Z0b_var, Z0b_varDefVal,
&      NumNodesNotDefault, NScreen, MyProc, NAbOut)
  ELSE
    SkipDataSet = .True.
  ENDIF
CASE("chezy_friction_coefficient_at_sea_floor")
  IF (LoadChezy) THEN
    CALL LoadAttrVec(Chezy, ChezyDefVal,
&      NumNodesNotDefault, NScreen, MyProc, NAbOut)
  ELSE
    SkipDataSet = .True.
  ENDIF
CASE("sea_surface_height_above_geoid")
  IF (LoadGeoidOffset) THEN
    CALL LoadAttrVec(GeoidOffset, GeoidOffsetDefVal,
&      NumNodesNotDefault, NScreen, MyProc, NAbOut)
  ELSE
    SkipDataSet = .True.
  ENDIF
CASE
& ("average_horizontal_eddy_viscosity_in_sea_water_wrt_depth")
  IF (LoadEVM) THEN
    CALL LoadAttrVec(EVM, EVMDefVal,
&      NumNodesNotDefault, NScreen, MyProc, NAbOut)
  ELSE
    SkipDataSet = .True.
  ENDIF
CASE
& ("average_horizontal_eddy_diffusivity_in_sea_water_wrt_depth")
  IF (LoadEVC) THEN
    CALL LoadAttrVec(EVC, EVCDefVal,

```

```

&          NumNodesNotDefault, NScreen, MyProc, NAbOut)
      ELSE
        SkipDataSet = .True.
      ENDIF
    CASE
&    ("min_and_max_primitive_weighting_in_continuity_equation")
      IF (LoadTau0MinMax) THEN
        CALL LoadAttrMat(Tau0MinMax, Tau0MinMaxNoOfVals,
&          Tau0MinMaxDefVal, NumNodesNotDefault,
&          NScreen, MyProc, NAbOut)
      ELSE
        SkipDataSet = .True.
      ENDIF
Casey 100210: Allow SWAN to handle wave refraction as a nodal attribute.
    CASE("wave_refraction_in_swan")
      IF (LoadSwanWaveRefrac) THEN
        CALL LoadAttrVec(SwanWaveRefrac, SwanWaveRefracDefVal,
&          NumNodesNotDefault, NScreen, MyProc, NAbOut)
      ELSE
        SkipDataSet = .TRUE.
      ENDIF
Corbitt 120321: Allow Advection to be handled locally instead of globally.
    CASE("advection_state")
      IF (LoadAdvectionState) THEN
        CALL LoadAttrVec(AdvectionState,AdvectionStateDefVal,
&          NumNodesNotDefault, NScreen, MyProc, NAbOut)
      ELSE
        SkipDataSet = .TRUE.
      ENDIF
    CASE("elemental_slope_limiter")
      IF( LoadEleSlopeLim ) THEN
        CALL LoadAttrVec(elemental_slope_limiter_grad_max,
&          EleSlopeLimDefVal, NumNodesNotDefault, NScreen,
&          MyProc, NAbOut)
      ELSE
        SkipDataSet = .True.
      ENDIF
    CASE DEFAULT
      SkipDataSet = .True.
      WRITE(16,1001)      ! Nodal Attributes file
      WRITE(16,1021) AttrName ! contains invalid name
      IF (NScreen.ne.0.and.MyProc.eq.0) THEN
        WRITE(ScreenUnit,1001)
        WRITE(ScreenUnit,1021) AttrName
      ENDIF
    END SELECT
    IF (SkipDataSet) THEN
      DO L=1, NumNodesNotDefault

```

```

        READ(13,*) Skipped
    END DO
    WRITE(16,'(9X,A8)') 'Skipped.'
    SkipDataSet = .False.
ELSE
    WRITE(16,'(/,9X,A18,A80)') 'Finished loading ', AttrName
ENDIF
END DO
C
1000 FORMAT('ERROR: The Model Parameter File (unit 15)')
1001 FORMAT('ERROR: The Nodal Attributes File (unit 13)')
1002 FORMAT('ERROR: The legacy StartDry File (unit 12)')
1003 FORMAT('ERROR: Spatially Varying Fric. Coeff. File (unit 21)')
C
1005 FORMAT('exists but cannot be opened.')
1011 FORMAT('was not found.')
1021 FORMAT('contains invalid name: ',A80)
9972 FORMAT(////,1X,'!!!!!!!!!!!! INPUT ERROR !!!!!!!!!',/)
9973 FORMAT(/,1X,'!!!!!!! EXECUTION WILL NOW BE TERMINATED !!!!!!!',/)
C
    RETURN
C
    -----
END SUBROUTINE ReadNodalAttr
C
    -----

C
C      S U B R O U T I N E      L O A D   A T T R   V E C
C
C
C      jg46.00 Subroutine to set a single set of nodal attributes to
C      their user-specified default values, then read the nondefault
C      values from the Nodal Attributes File (unit 13). This subroutine
C      is used for nodal attributes with only one value per node, hence
C      the suffix "vec" in the name.
C
C
C      -----
SUBROUTINE LoadAttrVec(AttributeData, Default, NumNodesNotDef,
&    NScreen, MyProc, NABOut)
    IMPLICIT NONE
    REAL(SZ), intent(out), dimension(NumOfNodes) :: AttributeData
    REAL(SZ), intent(in):: Default ! default value for all nodes
    INTEGER, intent(in) :: NumNodesNotDef ! number of nodes specified in
file
    INTEGER, intent(in) :: NScreen ! 1 for debug info to screen (unit 6)
    INTEGER, intent(in) :: MyProc ! in parallel, only MyProc=0 i/o to
screen
    INTEGER, intent(in) :: NABOut ! 1 to abbrev. output to unit 16

```



```

C      INTEGER NodeNum          ! node number listed in the file
C
C      Set all values to user-specified default values.
C      IF (NABOUT.EQ.0) WRITE(16,1001) Default
C      DO i=1, NumOfNodes
C          AttributeData(i) = Default
C      END DO
C
C      IF (NABOUT.EQ.0) WRITE(16,1005)
C      DO i=1, NumNodesNotDef
C          READ(13,*) NodeNum, AttributeData(NodeNum)
C          IF (NABOUT.EQ.0)
C      &      WRITE(16,1010) NodeNum, AttributeData(NodeNum)
C      END DO
C
C      1001 FORMAT(/,10X,'Set all nodes to the default value of ',E16.8,/)
C      1005 FORMAT(/,10X,'Now setting the following nodes to these values:',
C      &      /,10X,'NODE',5X,'DATA',5X/)
C      1010 FORMAT(7X,I8,6X,E16.8)
C
C      RETURN
C
C      -----
C      END SUBROUTINE LoadAttrVec
C      -----
C
C      -----
C      S U B R O U T I N E      L O A D   A T T R   M A T
C      -----
C
C      jgf46.00 Subroutine to load a single set of nodal attributes from
C      the Nodal Attributes File (unit 13) if there is more than one
C      value per node.
C
C      -----
C      SUBROUTINE LoadAttrMat(AttributeData, NumCol, Default,
C      &      NumNodesNotDef, NScreen, MyProc, NABOut)
C      IMPLICIT NONE
C      INTEGER, intent(in) :: NumCol ! number of columns in the matrix
C      REAL(SZ), intent(out),
C      &      dimension(NumOfNodes,NumCol) :: AttributeData
C      REAL(SZ), intent(in), dimension(NumCol) :: Default ! default values
C      INTEGER, intent(in) :: NumNodesNotDef ! number of nodes spec. in file
C      INTEGER, intent(in) :: NScreen ! 1 for debug info to screen (unit 6)
C      INTEGER, intent(in) :: MyProc ! in parallel, only MyProc=0 i/o to
screen
      INTEGER, intent(in) :: NABOut ! 1 to abbrev. output to unit 16

```

```

C      INTEGER NodeNum          ! node number listed in the file
C
C      Set all nodes to user-specified default values.
C      IF (NABOUT.EQ.0) WRITE(16,1001)
C      DO i=1, NumOfNodes
C          DO j=1, NumCol
C              AttributeData(i,j)=Default(j)
C          END DO
C      END DO
C
C      IF (NABOUT.EQ.0) WRITE(16,1005)
C      DO i=1, NumNodesNotDef
C          READ(13,*) NodeNum, (AttributeData(NodeNum,j),j=1,NumCol)
C          IF (NABOUT.EQ.0) WRITE(16,1010) NodeNum,
C      &      (AttributeData(NodeNum,j),j=1,NumCol)
C      END DO
C
C      1001 FORMAT(/,10X,'Set all nodes to the default values of ',/,
C      &      99E16.8,/)
C      1005 FORMAT(/,10X,'Now setting the following nodes to these values:',
C      &      /,10X,'NODE',5X,'DATA',5X/)
C      1010 FORMAT(7X,I6,6X,12(1X,E16.8))
C
C      RETURN
C
C      -----
C      END SUBROUTINE LoadAttrMat
C      -----
C
C      -----
C      S U B R O U T I N E      I N I T   N O D A L   A T T R
C      -----
C
C      jgf46.00 Subroutine to initialize and error check the nodal
C      attributes read in from the Nodal Attributes File (unit 13).
C
C      -----
C      SUBROUTINE InitNodalAttr(DP, NP, G, NScreen, ScreenUnit,
C      &      MyProc, NAbOut)
C      USE GLOBAL, ONLY : C3D, C2DDI
C      USE GLOBAL_3DVS, ONLY : Z0B
C      IMPLICIT NONE
C      INTEGER, intent(in) :: NP ! number of nodes in the grid file
C      REAL(SZ), intent(in), dimension(NP) :: DP ! array of bathymetric depths
C      REAL(SZ), intent(in):: G ! gravitational acceleration
C      INTEGER, intent(in) :: NScreen ! nonzero for debug info to screen

```

```

      INTEGER, intent(in) :: ScreenUnit ! i/o for debug info to screen
      INTEGER, intent(in) :: MyProc ! in parallel, only MyProc=0 i/o to
screen
      INTEGER, intent(in) :: NAbOut ! 1 to abbrev. output to unit 16
      INTEGER Tau0Dig1 ! determines the tau0 scheme
      INTEGER Tau0Dig2 ! determines whether tau0 is being output

      IF (Tau0.lt.0) THEN
        Tau0Dig1 = INT(Tau0) ! jgf47.30 truncate the fractional part
        !jgf47.34 round away from zero by subtracting 0.5d0
        Tau0Dig2 = INT( (Tau0 - REAL(Tau0Dig1))*10.0d0 - 0.5d0)
      ELSE
        Tau0Dig1 = 0
        Tau0Dig2 = 0
      ENDIF

C      ERROR CHECK: If a nodal attributes file is being used, check to
C      see that the number of nodes in the nodal attribute file is the
C      same as the number of nodes in the grid file.
      IF (NWP.NE.0.AND.NumOfNodes.NE.NP) THEN
        IF (NSCREEN.NE.0.AND.MYPROC.EQ.0) WRITE(ScreenUnit,9900)
        WRITE(16,9900)
9900      FORMAT(////,1X,'!!!!!!!!!!!! FATAL ERROR !!!!!!!!!',
&          //,1X,'The number of nodes in the grid file (unit 14) and'
&          //,1X,'the nodal attributes file (unit 13) must match.',
&          //,1X,'!!!!!!!! EXECUTION WILL NOW BE TERMINATED !!!!!',//)
        STOP ! We're toast.
      ENDIF

C
C      ERROR CHECK: If Chezy, Manning's or Quadratic friction was loaded
C      from the nodal attributes file, NOLIBF must be 1.
      IF ((LoadChezy.or.LoadManningsN.or.LoadQuadraticFric).and.
&      NoLiBF.ne.1) THEN
        IF (NSCREEN.NE.0.AND.MYPROC.EQ.0) WRITE(ScreenUnit,9800)
        WRITE(16,9800)
9800      FORMAT(////,1X,'!!!!!!!!!!!! FATAL ERROR !!!!!!!!!',
&          //,1X,'Nonlinear bottom friction coefficients were loaded'
&          //,1X,'from the nodal attributes file (unit 13), so ',
&          //,1X,'NoLiBF must be set to 1. It is set to ',i2,' in',
&          //,1X,'the model parameter (unit 15) file.',
&          //,1X,'!!!!!!!! EXECUTION WILL NOW BE TERMINATED !!!!!',//)
        STOP ! We're toast.
      ENDIF

C
C      ERROR CHECK: If Tau0=-3.x or -6.x in fort.15, then tau0 MUST be loaded
C      from nodal attributes file.
      IF ( ((Tau0Dig1.eq.-3).or.(Tau0Dig1.eq.-6))
&      .and.(.not.LoadTau0) ) THEN

```

```

        IF(NSCREEN.NE.0.AND.MYPROC.EQ.0) WRITE(ScreenUnit,9700) Tau0
        WRITE(16,9700)
9700    FORMAT(////,1X,'!!!!!!!!!! FATAL ERROR !!!!!!!!!',
&        //,1X,'Spatially and temporally varying tau0 was '
&        //,1X,'specified in the fort.15 file with Tau0=',E9.2,
&        //,1X,'but the base value was not specified in the '
&        //,1X,'nodal attributes file (unit 13). Please ',
&        //,1X,'load the base value using',
&        //,1X,'primitive_weighting_in_continuity_equation.',
&        //,1X,'!!!!!!!! EXECUTION WILL NOW BE TERMINATED !!!!!',//)
        STOP                                ! We're toast.
    ENDIF
C
C    I N I T      S T A R T D R Y
    IF (.not.LoadStartDry) THEN
        IF (NWP.eq.0) THEN
            ALLOCATE(STARTDRY(NP))
        ENDIF
        DO I=1, NP
            STARTDRY(I) = 0.0D0
        ENDDO
    ENDIF
C
C    I N I T      T A U 0
    IF (NWP.eq.0) THEN
        ALLOCATE(TAU0VAR(NP),TAU0BASE(NP))
        ALLOCATE(Tau0MinMax(NP,Tau0MinMaxNoOfVals))
    ENDIF
C
C    jgf46.25 If input tau0 is positive, set all nodes to that value.
    IF (.not.LoadTau0) THEN
        IF (Tau0.ge.0) THEN
            DO I=1,NP
                Tau0Var(I)=Tau0
            END DO
            WRITE(16,7) Tau0
7        FORMAT(/,5X,
&            'A SPATIALLY CONSTANT WEIGHTING COEFFICIENT (Tau0)'
&            //,5X,' WILL BE USED IN THE GENERALIZED WAVE',
&            ' CONTINUITY EQUATION.',
&            //,5X,'Tau0 = ',E15.8,2X,'1/sec',/)
        ELSE
C            If input tau0 is negative, set value using hardcoded scheme
C            based on depth
            DO I=1,NP
                Tau0Var(I)=Tau0NodalValue(Tau0,DP(I))
            ENDDO
            ! jgf47.30.TODO: This logging needs to be cleaned up.

```

```

        IF(Tau0.eq.-2) THEN
            WRITE(16,6) ! spatially vary tau0 according to hard coded
scheme
            WRITE(16,62) ! description of scheme
        62      FORMAT(/,5X,'IF DEPTH > 200          Tau0 = 0.005',
&              /,5X,'IF 200 > DEPTH > 1          Tau0 = 1/DEPTH ',
&              /,5X,'IF 1 > DEPTH          Tau0 = 1.0 ')
            ENDIF
            IF (.not.( (Tau0Dig1.eq.-3) .or. (Tau0Dig1.eq.-5) ) ) THEN
                WRITE(16,6) ! spatially vary tau0 according to hard coded
scheme
                WRITE(16,61) ! description of scheme
        61      FORMAT(/,5X,' IF DEPTH GE 10          -> TAU0 = 0.005',
&              /,5X,' IF DEPTH LT 10          -> TAU0 = 0.020',/)
            ENDIF
        ENDIF
    ENDIF
    FORMAT(/,5X,'A SPATIALLY VARIABLE WEIGHTING COEFFICIENT (Tau0)'
&      ,/,5X,' WILL BE USED IN THE GENERALIZED WAVE',
&      ' CONTINUITY EQUATION.',
&      /,5X,'THIS VALUE WILL BE DETERMINED AS FOLLOWS:')
C
C   jgf46.27 If we have already loaded the tau0 values directly from
C   the nodal attributes file, check to see if the default value was
C   negative. If so, this indicates that nodal values of tau0 that
C   were not explicitly set in the nodal attributes file should be set
C   according to one of the hard-coded tau0 schemes.
    IF (LoadTau0.and.Tau0DefVal.lt.0) THEN
        DO I=1,NP
            IF (Tau0Var(I).lt.0) THEN
                Tau0Var(I)=Tau0NodalValue(Tau0DefVal,DP(I))
            ENDIF
        ENDDO
    ENDIF

C
C   jgf47.06 Activate time varying tau0 and output tau0 if these options
C   were selected.

C   jgf47.30 Use Joannes' scheme for steady Tau0 in deep water and
C   other coarsely gridded areas, and time varying Tau0 in high
C   resolution areas.
    IF ( (Tau0Dig1.eq.-3).or.(Tau0Dig1.eq.-6)
&      .or. (Tau0Dig1.eq.-7) ) THEN
        HighResTimeVaryingTau0 = .True.
        DO I=1, NP
            Tau0Base(I) = Tau0Var(I)
        ENDDO
    ENDIF

```

```

C
C   jgf47.11 Also allow the min and max tau0 to be set from the fort.15,
C   bypassing the use of the fort.13 file for this purpose.
C   jgf47.30 Changed to emphasize full domain time varying tau0
C   IF ( Tau0Dig1.eq.-5 ) THEN
C       FullDomainTimeVaryingTau0 = .True.
C       IF ( .not.LoadTau0MinMax ) THEN
C           DO I=1, NP
C               Tau0MinMax(I,1) = Tau0FullDomainMin
C               Tau0MinMax(I,2) = Tau0FullDomainMax
C           ENDDO
C       ENDIF
C   ENDIF
C
C   jgf47.30: Output of tau0 is now activated by having a 0.1 fraction
C   for tau0
C   IF ( Tau0Dig2.eq.-1 ) THEN
C       OutputTau0 = .True.
C   ENDIF
C
C   jjw&sb46.38.sb01 If tau0 is loaded from nodal attributes file and
C   Tau0 is -3, time-varying tau0 optimizer will be applied in timestep.F
C   IF (HighResTimeVaryingTau0 .or. FullDomainTimeVaryingTau0) THEN
C       ALLOCATE(Tau0Temp(NP))
C       WRITE(16,8) ! jgf47.30.TODO: This logging should be consolidated.
C   ENDIF
8   FORMAT(/,5X,'A SPATIALLY TEMPORALLY VARIABLE OPTIMIZED '
&      ,/,5X,' WEIGHTING COEFFICIENT (Tau0) WILL BE USED '
&      ,/,5X,' IN THE GENERALIZED WAVE CONTINUITY EQUATION.',/)
C
C   jgf47.33 Enable time averaging of tau0 if requested.
C   IF (Tau0Dig1.eq.-6) THEN
C       TimeAveragedTau0 = .true.
C   ENDIF
C
C   jgf48.42 Enable back loaded time averaging of tau0 if requested.
C   IF (Tau0Dig1.eq.-7) THEN
C       BackLoadedTimeAveragedTau0 = .true.
C   ENDIF
C
C   jgf48.46 Allocate array to hold previous value tau0 for use in
C   time averaging, if necessary.
C   IF ( TimeAveragedTau0 .or. BackLoadedTimeAveragedTau0 ) THEN
C       ALLOCATE>LastTau0(NP))
C       DO I=1, NP
C           LastTau0(I) = Tau0Base(I)
C       ENDDO
C   ENDIF

```

```

C
C   I N I T   B O T T O M   F R I C T I O N
C   IF(NOLIBF.EQ.0) THEN
C       IFNLBF=0
C       IFLINBF=1
C       IFHYBF=0
C   ENDIF
C   IF(NOLIBF.EQ.1) THEN
C       IFNLBF=1
C       IFLINBF=0
C       IFHYBF=0
C   ENDIF
C   IF(NOLIBF.EQ.2) THEN
C       IFNLBF=0
C       IFLINBF=0
C       IFHYBF=1
C   ENDIF
C
C   Initialize bottom friction if it was not loaded from unit 13.
C   IF(C2DDI) THEN
C       IF((.not.LoadQuadraticFric).and.(.not.LoadManningsN).and.
&       (.not.LoadChezy)) THEN
C       IF (NoLiBF.eq.0) CF=Tau
C   If a nodal attributes file was read, FRIC was allocated there.
C       IF (NWP.eq.0) THEN
C           ALLOCATE(FRIC(NP))
C       ENDIF
C       DO I=1,NP
C           FRIC(I)=CF
C       END DO
C   ENDIF
C
C   jgf47.04 If a depth-dependent friction parameterization is used,
C   the value from the fort.15 file is used as a floor for the
C   minimum equivalent quadratic friction value.
C       IF (LoadManningsN) THEN
C           BFCdLLimit = CF
C       ENDIF
C   ENDIF
C
C   IF(C3D) THEN
C   Initialize 3D bottom roughness if it was not loaded from unit 13.
C       IF((.not.LoadZ0b_var)) THEN
C   If a nodal attributes file was read, Z0b_var was allocated there.
C       IF (NWP.eq.0) THEN
C           ALLOCATE(Z0b_var(NP))
C           ALLOCATE(FRIC(NP))
C       ENDIF

```

```

      DO I=1,NP
        Z0b_var(I)=Z0B
        FRIC(I)=CF
      END DO
    ENDIF
C   jgf47.04 If a depth-dependent friction parameterization is used,
C   the value from the fort.15 file is used as a floor for the
C   minimum equivalent quadratic friction value.
      IF (LoadZ0b_var.OR.LoadManningsN) THEN
        BFCdLLimit = CF
      ENDIF
    ENDIF
C
C   Initialize bridge pilings.
C
      IF (LoadBridgePilings) THEN
        DO I=1, NP
          IF (BridgePilings(I,1).ne.0) THEN ! only for nodes w/piers
            BridgePilings(I,3) = 4.d0 *
&              BridgePilings(I,3) / BridgePilings(I,4)
          ENDIF
        END DO
      ENDIF
C
C   I N I T   E D D Y   V I S C O S I T Y   &   D I F F U S I V I T Y
      IF (.not.LoadEVM) THEN
        IF (NWP.eq.0) THEN
          ALLOCATE(EVM(NP))
        ENDIF
        DO I=1,NP
          EVM(I)=ESLM
        END DO
      ENDIF
      IF (.not.LoadEVC.and.ESLC.ne.0) THEN
        IF (NWP.eq.0) THEN
          ALLOCATE(EVC(NP))
        ENDIF
        DO I=1,NP
          EVC(I)=ESLC
        END DO
      ENDIF
C
      RETURN
C   -----
C   END SUBROUTINE InitNodalAttr
C   -----

```



```

C -----
C   F U N C T I O N   T A U 0   N O D A L   V A L U E
C -----
C
C jgf46.27 Function to calculate tau0 based on the scheme selection
C and the depth. This assumes that Scheme is negative.
C
C -----
C   REAL(SZ) FUNCTION Tau0NodalValue(Scheme, Depth)
C   IMPLICIT NONE
C   REAL(SZ) Scheme
C   REAL(SZ) Depth
C
C   IF (Scheme.eq.-2.d0) THEN
C   Smoothly varying tau0 with depth.
C       IF(Depth.GE.200.) Tau0NodalValue=0.005
C       IF((Depth.LT.200.).AND.(Depth.GE.1.)) THEN
C           Tau0NodalValue=1./Depth
C       ENDIF
C       IF(Depth.LT.1.) Tau0NodalValue=1.0
C   ELSE
C   Abrupt variation in tau0 with depth.
C       IF(Depth.LE.10.) Tau0NodalValue=0.020d0
C       IF(Depth.GT.10.) Tau0NodalValue=0.005d0
C   ENDIF
C -----
C   END FUNCTION Tau0NodalValue
C -----
C
C -----
C   S U B R O U T I N E
C   C A L C U L A T E   T I M E   V A R Y I N G   T A U 0
C -----
C
C jgf47.08 Subroutine to calculate a new tau0 value. Called from
C GWCE_New in timestep.F each time the GWCE matrix is reset (i.e.,
C upon startup and whenever wetting and/or drying occurs in any
C subdomain. Based on Casey050711.
C
C -----
C   SUBROUTINE CalculateTimeVaryingTau0(TK, NNeigh, NeiTab, NP)
C   IMPLICIT NONE
C   REAL(SZ), intent(in) :: TK(:)          ! bottom friction
C   INTEGER, intent(in) :: NNeigh(:)       ! number of neighbor nodes
C   INTEGER, intent(in) :: NeiTab(:, :)    ! table of neighbor nodes
C   INTEGER, intent(in) :: NP              ! number of nodes in the domain
C   REAL(SZ) CaseySum ! sum of tau0temp values around a particular node

```

Casey 050711 : Made changes for averaged variable Tau0.

cjjw46.39.sb01 : "high/low LIMITED" variable G.

!jgf47.30: Distinction between fulldomain and hi res only

IF ( FullDomainTimeVaryingTau0 ) THEN

DO i = 1, NP

Tau0Temp(i)=Tau0MinMax(i,1)+1.5\*TK(i)

IF (Tau0Temp(i).lt.Tau0MinMax(i,1)) THEN

Tau0Temp(i)=Tau0MinMax(i,1)

ENDIF

IF(Tau0Temp(i).gt.Tau0MinMax(i,2)) THEN

Tau0Temp(i)=Tau0MinMax(i,2)

ENDIF

ENDDO

ENDIF

IF ( HighResTimeVaryingTau0 ) THEN

DO i = 1, NP

IF(Tau0Base(i).lt.0.025) THEN

Tau0Temp(i)=Tau0Base(i) ! not time varying

ELSE

Tau0Temp(i)=Tau0Base(i)+1.5\*TK(i) ! time varying

IF (Tau0Temp(i).gt.0.2) Tau0Temp(i)=0.2 ! ceiling

ENDIF

ENDDO

ENDIF

! smoothing

DO I=1, NP

CaseySum = 0.0

DO J=1,NNeigh(I)

CaseySum = CaseySum + Tau0Temp(NeiTab(I,J))

ENDDO

TAU0VAR(I) = CaseySum / NNeigh(I)

ENDDO

C

C jgf47.33 Perform time averaging of tau0 if requested.

IF (TimeAveragedTau0) THEN

DO I=1, NP

TAU0VAR(I) = 0.5d0\*TAU0VAR(I) + 0.5d0\*LastTau0(I)

LastTau0(I) = TAU0VAR(I)

ENDDO

ENDIF

C

C jgf48.42 Perform backloaded time averaging of tau0 if requested.

IF (BackLoadedTimeAveragedTau0) THEN

DO I=1, NP

TAU0VAR(I) = AlphaTau0\*TAU0VAR(I)

& + (1.d0-AlphaTau0)\*LastTau0(I)

LastTau0(I) = TAU0VAR(I)

```

      ENDDO
    ENDIF
C -----
END SUBROUTINE CalculateTimeVaryingTau0
C -----

C -----
C SUBROUTINE
C APPLY 2D BOTTOM FRICTION
C -----
C
C jgf46.00 Subroutine to apply 2D bottom friction from turbulent
C viscous effects as well as bridge pilings. This is used in the
C time stepping loop.
C -----
C
C sb46.28sb02 Lower limit of Cd was added as an argument.
C jgf47.04 Argument for lower limit of Cd was removed; this value
C is now specified by the user in fort.15.
C -----
C
SUBROUTINE Apply2DBottomFriction(NodeNumber, UU1, VV1, DP, ETA2, G,
& IFNLFA, NP, TK)
  USE SIZES
  IMPLICIT NONE
  INTEGER, intent(in) :: NodeNumber ! index of node under consideration
  INTEGER, intent(in) :: NP          ! number of nodes in grid
  REAL(SZ), intent(in), dimension(NP) :: UU1 ! x-dir velocities
  REAL(SZ), intent(in), dimension(NP) :: VV1 ! y-dir velocities
  REAL(SZ), intent(in), dimension(NP) :: DP ! bathymetric depths
  REAL(SZ), intent(in), dimension(NP) :: ETA2 ! water surf. elevations
  REAL(SZ), intent(in) :: G ! gravitational constant
  INTEGER, intent(in) :: IFNLFA ! nonlin. finite amp. flag
  REAL(SZ), intent(inout), dimension(NP) :: TK ! depth avg. fric.
C
  REAL(SZ) UV1 ! velocity magnitude (speed)
  REAL(SZ) H1 ! total depth
  REAL(SZ) Fr
  REAL(SZ) FricBP
  REAL(SZ) BK ! BK(1) is pier shape factor
  REAL(SZ) BALPHA ! BALPHA(2) is constriction fraction
  REAL(SZ) BDELX ! BDELX(3) is effective delx
  REAL(SZ) n01 ! ManningsN factor for a particular node, for particular
direction
  REAL(SZ) angle ! Flow direction

```

```

    INTEGER iflowdir    ! code for wind direction
C
C   compute direction that the flow is coming from
    if((UU1(I).eq.0).and.(VV1(I).eq.0))then
        angle=0.d0
    else
        angle=atan2(VV1(I),UU1(I))
    endif
    angle=360.*angle/(2*3.141592654d0)
    iflowdir=0
    if((angle.gt.-15.).and.(angle.le.15)) iflowdir=1
    if((angle.gt.15.).and.(angle.le.45)) iflowdir=2
    if((angle.gt.45.).and.(angle.le.75)) iflowdir=3
    if((angle.gt.75.).and.(angle.le.105)) iflowdir=4
    if((angle.gt.105.).and.(angle.le.135)) iflowdir=5
    if((angle.gt.135.).and.(angle.le.165)) iflowdir=6
    if((angle.gt.165.).and.(angle.le.180)) iflowdir=7
    if((angle.gt.-45.).and.(angle.le.-15)) iflowdir=12
    if((angle.gt.-75.).and.(angle.le.-45)) iflowdir=11
    if((angle.gt.-105.).and.(angle.le.-75)) iflowdir=10
    if((angle.gt.-135.).and.(angle.le.-105)) iflowdir=9
    if((angle.gt.-165.).and.(angle.le.-135)) iflowdir=8
    if((angle.ge.-180.).and.(angle.le.-165)) iflowdir=7
C
C   define roughness from values
    n01=z0land(NodeNumber,iflowdir)
C   Step 0. Convert Manning's N to Cd, if necessary.
    IF (LoadManningsN) THEN
        DO I=1, NP
            FRIC(I)=g*n01*ManningsN(I)**2.d0
            & /( ( DP(I)+IFNLFA*ETA2(I) )**(1.d0/3.d0) ) ! sb46.28sb02
            !sb46.28sb02 Lower limit is applied here.
            IF(FRIC(I).LT.BFCdLLimit) THEN
                FRIC(I) = BFCdLLimit
            ENDIF
        ENDDO
    ENDIF
C
C   ... Convert Chezy to Cd, if necessary.
    IF (LoadChezy) THEN
        DO I=1, NP
            FRIC(I)=G/(Chezy(I)**2)
        END DO
    ENDIF
C
C   Step 1. Apply friction arising from turbulent viscous interaction
C   with the sea floor.

```

```

DO I=1, NP
  UV1=SQRT(UU1(I)*UU1(I)+VV1(I)*VV1(I))
  H1=DP(I)+IFNLFA*ETA2(I)
  TK(I)= FRIC(I)*
&      ( IFLINBF +      ! linear
&      (UV1/H1) * (IFNLBF ! nonlinear
&      + IFHYBF*(n01**2)*(1+(HBREAK/H1)**FTHETA)**(FGAMMA/FTHETA))) !
hybrid
END DO

C
C   Step 2. Apply friction arising from flow interaction with bridge
C   pilings, if required.
IF (LoadBridgePilings) THEN
  DO I=1, NP
    UV1=SQRT(UU1(I)*UU1(I)+VV1(I)*VV1(I))
    H1=DP(I)+IFNLFA*ETA2(I)
    Fr=UV1*UV1/(G*H1)
    BK = BridgePilings(I,1)
    BALPHA = BridgePilings(I,2)
    BDELX = BridgePilings(I,3)
    FricBP=(H1/BDELX)*BK*(BK+5.d0*Fr*Fr-0.6d0)
&      *(BALPHA+15.d0*BALPHA**4)
    TK(I)=TK(I)+FricBP*UV1/H1
  END DO
ENDIF

C
C   RETURN
C   -----
END SUBROUTINE Apply2DBottomFriction
C   -----
C
C   S U B R O U T I N E
C   A P P L Y   3   D   B O T T O M   F R I C T I O N
C   -----
C
C   jgf46.00 Subroutine to apply 3D bottom friction from turbulent
C   viscous effects as well as bridge pilings. This is used in the
C   time stepping loop.
C   -----
C   -----
SUBROUTINE Apply3DBottomFriction(Q, SIGMA, DP, ETA2, G,
&   IFNLFA, NP, TK, NFEN)
  USE SIZES
  USE GLOBAL_3DVS, ONLY : Z0B
  IMPLICIT NONE
  INTEGER, intent(in) :: NP, NFEN          ! number of nodes in grid
Horizontal and Vertical
  COMPLEX(SZ), intent(in), dimension(NP,NFEN) :: Q ! x-dir velocities

```

```

REAL(SZ), intent(in), dimension(NFEN) :: SIGMA ! x-dir velocities
REAL(SZ), intent(in), dimension(NP) :: DP ! bathymetric depths
REAL(SZ), intent(in), dimension(NP) :: ETA2 ! water surf. elevations
REAL(SZ), intent(in) :: G ! gravitational constant
INTEGER, intent(in) :: IFNLFA ! nonlin. finite amp. flag
REAL(SZ), intent(inout), dimension(NP) :: TK! depth avg. fric.

```

C

```

INTEGER NH
REAL(SZ) Z0B1 ! velocity magnitude (speed)
REAL(SZ) UV1 ! velocity magnitude (speed)
REAL(SZ) H1 ! total depth
REAL(SZ) Fr
REAL(SZ) FricBP
REAL(SZ) BK ! BK(1) is pier shape factor
REAL(SZ) BALPHA! BALPHA(2) is constriction fraction
REAL(SZ) BDELX ! BDELX(3) is effective delx

```

C

C Determine the bottom roughness length either from fort.15, from Manning's n  
C or as read in from nodal attributes

```

DO NH=1,NP
  H1=DP(NH)+IFNLFA*ETA2(NH)
  IF (LoadZ0B_var) THEN
    Z0B1 = Z0B_var(NH)
  ELSEIF (LoadManningsN) THEN
    Z0B1 = ( H1 ) * exp(-(1.0D0+
&          ( (0.41D0*( H1 )**(1.0D0/6.0D0) ) /
&          (ManningsN(NH)*sqrt(g)) ) ))
  ELSE
    Z0B1 = Z0B
  ENDIF

  FRIC(NH)= (1.D0 / ( (1.D0/0.41D0) *
& LOG((ABS( ( ( SIGMA(2)-SIGMA(1) )/2.d0 ) *(H1) ) + Z0B1 )/Z0B1)
& ) )**2.D0

  TK(NH)= FRIC(NH) * ABS(Q(NH,1))

  IF (LoadBridgePilings) THEN
    Fr=ABS(Q(NH,1))*ABS(Q(NH,1))/(G*H1)
    BK = BridgePilings(I,1)
    BALPHA = BridgePilings(I,2)
    BDELX = BridgePilings(I,3)
    FricBP=(H1/BDELX)*BK*(BK+5.d0*Fr*Fr-0.6d0)
&      *(BALPHA+15.d0*BALPHA**4)
    TK(I)=TK(I)+FricBP*ABS(Q(NH,1))/H1
  ENDIF
ENDDO

```

C

```

      RETURN
C -----
      END SUBROUTINE Apply3DBottomFriction
C -----

C -----
C   S U B R O U T I N E
C   A P P L Y   D I R E C T I O N A L   W I N D   R E D U C T I O N
C -----
C
C   jgf46.00 Subroutine to calculate the land wind reduction factor
C   based on a table of directional wind drag values. Originally
C   written into the hstart.F file by jjw in jjw-42.06j. This is used
C   in hstart.F and timestep.F.
C
C   jgf49.1001 Extracted the application of the canopy coefficient and
C   placed in the ApplyCanopyCoefficient subroutine.
C -----
      SUBROUTINE ApplyDirectionalWindReduction(NodeNumber, WindDragCo,
&      WindMag, BathymetricDepth, Elevation, CutOffDepth, G,
&      WindX, WindY)
      USE SIZES
      IMPLICIT NONE
      INTEGER, intent(in) :: NodeNumber ! index of node under consideration
      REAL(SZ), intent(in) :: WindDragCo ! wind drag coefficient
      REAL(SZ), intent(in) :: WindMag ! wind magnitude
      REAL(SZ), intent(in) :: BathymetricDepth ! a.k.a. dp(i), depth below
geoid
      REAL(SZ), intent(in) :: Elevation ! a.k.a. eta2(i)
      REAL(SZ), intent(in) :: CutOffDepth ! a.k.a. h0, user-spec. min. depth
      REAL(SZ), intent(in) :: G ! gravitational constant

      REAL(SZ), intent(inout) :: WindX ! x-dir component of wind velocity
      REAL(SZ), intent(inout) :: WindY ! x-dir component of wind velocity

      REAL(SZ) z0m ! marine roughness coefficient based on Garratt's
formula
      REAL(SZ) angle ! direction wind is coming from
      INTEGER idir ! code for wind direction
      REAL(SZ) z0l ! drag for a particular node, for particular direction
      REAL(SZ) TotalDepth ! bathymetric depth + sea surface elevation
      REAL(SZ) fr ! land wind reduction factor

C
C   compute marine roughness coefficient based on Garratt's formula
      z0m=(0.018d0/G)*WindDragCo*WindMag**2.d0
C
C   compute direction that the wind is coming from

```

```

if((WindX.eq.0).and.(WindY.eq.0))then
  angle=0.d0
else
  angle=atan2(WindY,WindX)
endif
angle=360.*angle/(2*3.141592654d0)
idir=0
if((angle.gt.-15.).and.(angle.le.15)) idir=1
if((angle.gt.15.).and.(angle.le.45)) idir=2
if((angle.gt.45.).and.(angle.le.75)) idir=3
if((angle.gt.75.).and.(angle.le.105)) idir=4
if((angle.gt.105.).and.(angle.le.135)) idir=5
if((angle.gt.135.).and.(angle.le.165)) idir=6
if((angle.gt.165.).and.(angle.le.180)) idir=7
if((angle.gt.-45.).and.(angle.le.-15)) idir=12
if((angle.gt.-75.).and.(angle.le.-45)) idir=11
if((angle.gt.-105.).and.(angle.le.-75)) idir=10
if((angle.gt.-135.).and.(angle.le.-105)) idir=9
if((angle.gt.-165.).and.(angle.le.-135)) idir=8
if((angle.ge.-180.).and.(angle.le.-165)) idir=7
C
C   define land roughness from usace values
z0l=z0land(NodeNumber,idir)
C
C   reset z0l depending on situation
if(z0l.le.0.006) then
c   coe set their value to a marine value -> reset to correct marine value
  z0l=z0m
else
c   coe set their value to a land value -> proceed with checking this value
  TotalDepth = BathymetricDepth + Elevation
  if( (TotalDepth.gt.2*CutOffDepth).and.
&    (BathymetricDepth.lt.20)) then
c   compute adjusted z0l to account for overland flooding - do this only
c   in the case where the water column is greater than twice h0 and
c   you are not in a river (I assume that rivers are deeper than 20m
c   and have z0l>0.006)
    z0l=z0l-TotalDepth/30. ! correction for overland flooding
  endif
endif
C
c   compute land wind reduction factor
if(z0l.gt.0.0001) then
  fr=(z0m/z0l)**0.0706d0
else
  fr=1.000d0
endif
if(fr.gt.1.0000d0) fr=1.0000d0

```



```

c      adjust time interpolated wind field
      WindX = fr*WindX
      WindY = fr*WindY
C
      RETURN
C
      -----
      END SUBROUTINE ApplyDirectionalWindReduction
C
      -----

C      -----
C      S U B R O U T I N E
C      A P P L Y   C A N O P Y   C O E F F I C I E N T
C      -----
C      jgf49.1001 Subroutine to apply the canopy coefficient. Was
C      originally included in the subroutine ApplyDirectionalWindReduction;
C      this combination implicitly assumed that the two attributes would
C      always be used together.
C      -----
      SUBROUTINE ApplyCanopyCoefficient(NodeNumber, WindX, WindY)
      IMPLICIT NONE
      INTEGER, intent(in) :: NodeNumber ! index of node under consideration
      REAL(SZ), intent(inout) :: WindX ! x-dir component of wind velocity
      REAL(SZ), intent(inout) :: WindY ! x-dir component of wind velocity
C
      WindX = vcanopy(NodeNumber)*WindX
      WindY = vcanopy(NodeNumber)*WindY
C
      RETURN
C
      -----
      END SUBROUTINE ApplyCanopyCoefficient
C
      -----

C      -----
C      S U B R O U T I N E
C      R E A D   L E G A C Y   S T A R T   D R Y   F I L E
C      -----
C
C      jgf46.00 Subroutine to load up the legacy startdry file (unit
C      12). This is just a cut-and-paste from the section of the
C      READ_INPUT subroutine that did the same thing. This subroutine is
C      never called. It is vestigial and listed here purely as reference
C      material.
C
C      -----
      SUBROUTINE ReadLegacyStartDryFile(NP, NScreen, ScreenUnit,
&      MyProc, NAbOut)

```

```

      IMPLICIT NONE
      INTEGER, intent(in) :: NP ! number of nodes in grid file
      INTEGER, intent(in) :: NScreen ! nonzero for debug info to screen
      INTEGER, intent(in) :: ScreenUnit ! i/o for debug info to screen
      INTEGER, intent(in) :: MyProc ! in parallel, only MyProc=0 i/o to
screen
      INTEGER, intent(in) :: NAbOut ! 1 to abbrev. output to unit 16

      INTEGER JKI          ! node number from file
      INTEGER NE2          ! number of elements, according to fort.12 file
      INTEGER NP2          ! number of nodes, according to fort.12 file

      CHARACTER(len=80) AGRID2 ! users comment/description line
      REAL(SZ) DUM1, DUM2 ! data that we want to skip

      OPEN(12,FILE=TRIM(INPUTDIR)//'/'/'fort.12')
C
C...  READ STARTDRY INFORMATION FROM UNIT 12
      READ(12,'(A80)') AGRID2
      WRITE(16,2038) AGRID2
2038  FORMAT(5X,'STARTDRY FILE IDENTIFICATION : ',A80,/)
      READ(12,*) NE2,NP2
C
C...  CHECK THAT NE2 AND NP2 MATCH WITH GRID FILE
C      IF((NE2.NE.NE).OR.(NP2.NE.NP)) THEN
      IF(NP2.NE.NP) THEN
          IF(NSCREEN.NE.0.AND.MYPROC.EQ.0) WRITE(ScreenUnit,9900)
          WRITE(16,9900)
9900  FORMAT(////,1X,'!!!!!!!!!!!! FATAL ERROR !!!!!!!!!!!',
&          //,1X,'THE PARAMETER NE2 AND NP2 MUST MATCH NE AND NP ',
&          /,1X,'USER MUST CHECK FORT.12 INPUT FILE ',
&          //,1X,'!!!!!!!! EXECUTION WILL NOW BE TERMINATED !!!!!!!',//)
          STOP
      ENDIF
C
C...  READ IN STARTDRY CODE VALUES
      DO I=1,NP
          READ(12,*) JKI,DUM1,DUM2,STARTDRY(JKI)
          IF(JKI.NE.I) THEN
              IF(NSCREEN.NE.0.AND.MYPROC.EQ.0) WRITE(ScreenUnit,99805)
              WRITE(16,99805)
99805  FORMAT(////,1X,'!!!!!!!!!!!! WARNING - NONFATAL ',
&          'INPUT ERROR !!!!!!!!!!!',
&          //,1X,'YOUR NODE NUMBERING IS NOT SEQUENTIAL ',
&          'CHECK YOUR UNIT 12 INPUT FILE CAREFULLY',//)
          ENDIF
      END DO
C

```

```

C...  CLOSE UNIT 12 FILE
      CLOSE(12)
C
      RETURN
C
      -----
      END SUBROUTINE ReadLegacyStartDryFile
C
      -----

C
C          S U B R O U T I N E
C      R E A D   L E G A C Y   B O T T O M   F R I C T I O N   F I L E
C
C      -----
C
C      jgf46.00 Subroutine to load up the legacy Spatially Varying
C      Friction Coefficient File (unit 21). This is just a cut-and-paste
C      from the section of the READ_INPUT subroutine that did the same
C      thing. This subroutine is never called. It is vestigial and listed
C      here purely as reference material.
C
C      -----
C
      SUBROUTINE ReadLegacyBottomFrictionFile(NP, NScreen, ScreenUnit,
&      MyProc, NAbOut)
      IMPLICIT NONE
      INTEGER, intent(in) :: NP ! number of nodes in grid file
      INTEGER, intent(in) :: NScreen ! nonzero for debug info to screen
      INTEGER, intent(in) :: ScreenUnit ! i/o for debug info to screen
      INTEGER, intent(in) :: MyProc ! in parallel, only MyProc=0 i/o to
screen
      INTEGER, intent(in) :: NAbOut ! 1 to abbrev. output to unit 16

      CHARACTER(len=80) AFRIC ! user's comment/description line
      INTEGER NHG ! node number from file

      OPEN(21,FILE=TRIM(INPUTDIR)//'/'/'fort.21')
      READ(21,'(A80)') AFRIC
      DO I=1,NP
        READ(21,*) NHG,FRIC(NHG)
        IF(NHG.NE.I) THEN
          IF(NSCREEN.NE.0.AND.MYPROC.EQ.0) WRITE(ScreenUnit,99803)
          WRITE(16,99803)
99803      FORMAT(////,1X,'!!!!!!!!!!!! WARNING - FATAL ',
&          'INPUT ERROR !!!!!!!!!!!!!',//,1X,
&          'YOUR NODAL FRICTION NUMBERING IS NOT SEQUENTIAL ',
&          '/',1X,'CHECK YOUR UNIT 21 INPUT FILE CAREFULLY',//,1X,
&          '!!!!!!!! EXECUTION WILL NOW BE TERMINATED !!!!!!!',//)
          STOP
        ENDIF
      ENDIF

```

```

      END DO
      WRITE(16,3601) AFRIC
3601  FORMAT(/,5X,'FRICTION FILE IDENTIFICATN : ',A80,/)
      IF(NABOUT.NE.1) THEN
        WRITE(16,2080)
2080  FORMAT(/,10X,'NODE',5X,'BOTTOM FRICTION FRIC',5X,/)
        DO I=1,NP
          WRITE(16,2087) I,FRIC(I)
2087  FORMAT(7X,I6,6X,E17.10)
        END DO
      ELSE
        WRITE(16,3504)
3504  FORMAT(/,5X,'NODAL BOTTOM FRICTION VALUES ARE AVAILABLE',
&      /,6X,' IN UNIT 21 INPUT FILE')
      ENDIF
C
      RETURN
C  -----
C  END SUBROUTINE ReadLegacyBottomFrictionFile
C  -----

C-----
C      END MODULE NodalAttributes
C-----
C-----

```

## VITA

### Personal Information

Name: Henok Kefelegn Demissie

### Education

B.Sc., Civil Engineering, 2004, Addis Ababa University, Addis Ababa, Ethiopia

M.Sc., Civil/Structural Engineering, 2007, Addis Ababa University, Addis Ababa, Ethiopia

M.Sc., Civil/Coastal Engineering, 2015, University of North Florida, Jacksonville, Florida

### Honors and Awards

- Delores A. Auzenne Fellowship, State-funded Scholarship 2014-2015
- Taylor Engineering Endowed Professorship Scholarship 2013-2015
- Super Volunteer Award, Jacksonville African Community Organization (JACO) 2014

### Association Memberships

- National Society of Leadership and Success and Coasts
- Oceans, Ports and Rivers Institute (COPRI)
- American Society of Civil Engineers

### Professional Experience

University of North Florida Jacksonville, Florida

- Graduate Research Assistant September 2013- present

#### Inspection Depot Jacksonville, Florida

- Quality Assurance Representative March 2012- August 2013
- Audited Uniform Mitigation Verification Inspection reports, managed disputes
- Developed Training materials ([www.ciaconnect.com/education/course-categories](http://www.ciaconnect.com/education/course-categories))

#### Alfred Talke Logistic Service Qatar Doha, Qatar

- Project Engineer September 2010-April 2011
- Coordinated all technical activities on assigned projects
- Monitored work for compliance to applicable codes, accepted engineering practices and standards

#### Iberdrola Engineering and Construction Qatar Doha, Qatar

- Civil Supervisor April 2007-June 2010
- Inspected entire integrated underground systems scope, concrete foundations
- Supervised Inspectors and ensured they are fully familiarized with project requirements
- Ensured design requirements are met in compliance with drawings, specifications and applicable codes
- Monitored and verified all testing within the discipline
- Worked in close cooperation with the Completions Group, closed punch list items

#### Professional Presentation

- This thesis “Bottom Friction Assessment for Hydrodynamic Currents in the Lower St. Johns River ” was presented to professional audience, at the 2015 ADCIRC users group meeting, National Weather Service Climate Prediction Center, College Park, Maryland, 03/31/2015